T · H · E
# OHIO STATE UNIVERSITY

MICROWAVE ANTENNA TECHNOLOGY
Final Report

OSU Reflector Antenna Code - User's Manual

T.H. Lee
R.C. Rudduck

The Ohio State University

# ElectroScience Laboratory

Department of Electrical Engineering
Columbus, Ohio   43212

Final Report 717822-3

19960503 066

DTIC QUALITY INSPECTED 1

NOTICES


When Government drawings, specifications, or other data are
used for any purpose other than in connection with a definitely
related Government procurement operation, the United States
Government thereby incurs no responsibility nor any obligation
whatsoever, and the fact that the Government may have formulated,
furnished, or in any way supplied the said drawings, specifications,
or other data, is not to be regarded by implication or otherwise as
in any manner licensing the holder or any other person or corporation,
or conveying any rights or permission to manufacture, use, or sell
any patented invention that may in any way be related thereto.

50272-101

| REPORT DOCUMENTATION PAGE | 1. REPORT NO. | 2. | 3. Recipient's Accession No. |
|---|---|---|---|

**4. Title and Subtitle**
MICROWAVE ANTENNA TECHNOLOGY
: OSU Reflector Antenna Code -- User's Manual

**5. Report Date**
May 1987

**6.**

**7. Author(s)**
T.H. Lee, R.C. Rudduck

**8. Performing Organization Rept. No.**
717822-3

**9. Performing Organization Name and Address**
The Ohio State University ElectroScience Laboratory
1320 Kinnear Road
Columbus, Ohio 43212

**10. Project/Task/Work Unit No.**

**11. Contract(C) or Grant(G) No**
(C
(G.

**12. Sponsoring Organization Name and Address**

**13. Type of Report & Period Covered**
Final

**14.**

**15. Supplementary Notes**

**16. Abstract (Limit: 200 words)**

The major purpose of this research is to provide a computer-aided analysis and design capability for microwave reflector antenna systems, in the 1-40 GHz range. This capability will allow for the prediction of antenna gain and antenna temperature performance of reflector antennas, under various atmospheric conditions.

The analysis and design capability was accomplished in part by further development of the OSU Reflector Antenna Code. The analysis capability of the Reflector Code was used to guide the design of the seven reflector antennas, of both the focal-point and Cassegrain types. These seven reflector antenna designs were fabricated and tested. The measured data obtained from these tests were used to validate the Reflector Antenna Code. The use of the Reflector Antenna code is documented in Volume III of this final report.

This report documents the seven reflector antenna designs which were fabricated and measured to demonstrate and test the computer-aided analysis and design capability. Selected pattern and gain data calculated by the Reflector Antenna Code and the validating measured data are given. A more comprehensive set of measured data for the seven reflector antenna designs is presented in Volume II of this final report.

**17. Document Analysis  a. Descriptors**

**b. Identifiers/Open-Ended Terms**

**c. COSATI Field/Group**

Der AQ U 95-1391

| 19. Security Class (This Report) Unclassified | 21. No. of Pages 377 |
|---|---|
| 20. Security Class (This Page) Unclassified | 22. Price |

(See ANSI-Z39.18)  See Instructions on Reverse

OPTIONAL FORM 272 (4-77)
(Formerly NTIS-35)
Department of Commerce

# TABLE OF CONTENTS

## LIST OF COMMANDS

# LIST OF TABLES

# LIST OF FIGURES

**Figure**

Page

xi

xiii

# 1. INTRODUCTION

The OSU Reflector Antenna Code is an updated version of the NEC Reflector Antenna Code [1,2]. The NEC Reflector Antenna Code has the capability for both near-field and far-field computations for reflector antennas with paraboloidal surfaces. A key feature of the code is its capability for a general reflector rim shape. Many new capabilities have been added in this updated version of the OSU Reflector Antenna Code. This report documents all the capabilities and describes the operation of the updated version of the Reflector Antenna Code.

The theoretical approach for computing the fields of the reflector antenna is based on a combination of Geometrical Theory of Diffraction (GTD) and Aperture Integration (AI) techniques. Typically, AI, also known as the Aperture Field Method, is used to compute the main beam and near sidelobes; GTD is used to compute the wide-angle sidelobes and backlobes. For near field calculations, GTD is sometimes used for the whole region including the near axis region if the near field points are close to the aperture. The GTD and AI approaches used for the reflector code have a basic limitation on the minimum-size reflector that can be modeled. By comparison to the exact solution for a circular disk the code has been shown to be reasonably accurate for reflector diameters as small as 3 wavelengths [3]. There is no limitation on the maximum size of the reflector for the basic analysis. Two efficient techniques have been employed to carry out the aperture integration which is performed over the portion of the aperture plane inside the reflector rim. One is the large subaperture method; the other is the rotating grid method. The subapertures can be electrically large, thus minimizing the computer storage and also the amount of numerical integration required. The major feature of the rotating grid method is that the y-integrations are carried out for each column of the aperture and each one-dimensional integration result is stored. The stored values for the y-integrations are then used for each pattern angle in the plane perpendicular to the y-axis, thus the efficiency approaches that of a one-dimensional integration. Computational advantages on the order of 10 to 100 in computer time ratios are frequently encountered because of the rotating grid feature.

Two GTD approaches have been applied to analyze the wide-angle sidelobes and near field patterns for reflector antennas. The first method is called "segmented-rim GTD" which has been used in the NEC Reflector Antenna Code. The second method is called "multi-point GTD" which is a new technique to calculate the diffracted field. The segmented-rim GTD analysis of the reflector is similar to that of diffraction by a flat plate [4] in that the reflector rim is divided into segments and each segment is treated as an edge of a flat plate which is tangent to the reflector surface. Uniform GTD techniques [5] are used to calculate the edge diffracted field and the slope diffracted field which are the same as those for infinite straight edges. The use of the corner diffraction solution proposed by Burnside et al. [6]

permits the reflector rim to be modeled by piece-wise linear segments. Consequently, reflector rim shapes with corners, such as square or rectangular, can be analyzed by GTD. Smooth rim shapes, such as elliptical or circular, are also modeled by segmenting the reflector rim. Thus, this approach has the added advantage that it eliminates the need for special techniques, such as equivalent line source calculations, for caustic regions near the rear axis as is required by two-point GTD techniques. The limitations of this rim segmentation approach have been investigated for circular rim shapes [3]; and accurate results are achieved if the rim segments are sufficiently small. Typically, 36 rim segments are sufficient to calculate far field patterns for reflector diameters smaller than 22 wavelengths. For reflector diameters of D=100 wavelengths and 1000 wavelengths the required number of rim segments is usually about 80 and 240, respectively.

The multi-point GTD method [7] applies numerical techniques to find the diffraction points and caustic distances. For reflector antennas with corners, corner diffracted fields are added to the multi-point GTD. The reflector rim is also divided into many small rim segments, and the geometry associated with these segments such as the unit vectors are calculated and stored. First, the "true corner" is found by checking the angles formed by two adjacent segments. Then the diffraction points are searched around the reflector rim. The diffracted fields from the diffraction points and corners are then calculated and added. The multi-point GTD method greatly improves the computation time compared to the segmented rim GTD since the latter method includes the contributions from all the rim segments while the multi-point GTD only includes the contributions from true diffraction points and corners. The disadvantage of the multi-point GTD method is that it can not be used when the field point is near or at the caustic of the reflector, because a very large number of diffraction points can be found. In this case, the multi-point GTD method must be supplemented by the segmented-rim GTD method.

The conventional aperture integration (AIC) method used in the NEC Reflector Antenna Code approximates the aperture fields by the geometrical optics (G.O.) fields, i.e., the reflected fields from the reflector surface. Thus, the integration is performed only over the projected aperture of the reflector. The G.O. approximation is good enough for most cases. However, when the effect of the diffracted field contribution to the aperture fields become significant, the G.O. field approximation is no longer adequate. An extended aperture integration (AIE) method has been developed and included in the latest version of the Reflector Antenna Code to correct the inadequacy of the G.O. approximation of aperture fields. In the AIE method, the aperture is extended to include the diffracted fields outside the original projected G.O. aperture. This method improves the results of aperture integration especially in the wider sidelobe regions. However, without the multi-point GTD, the AIE method will not be as efficient since excessive

1-2

CPU time will be needed to calculate the corrected aperture fields by the method of segmented-rim GTD.

Feed blockage is modeled by the physical optics model of a rectangular or a circular flat plate [8] in the NEC Reflector Antenna Code. These models should be valid in the main beam and near sidelobe region where the feed blockage effect is most significant, because the physical optics model is accurate for forward scattering. However, these feed blockage models have to be inside the G.O. region of the reflector in order to be accurate. Thus, feed scattering effects cannot be modeled this way for an offset reflector. Consequently, the AIE method is used to calculate feed scattering effects for offset reflector antennas.

Strut scattering is modeled by assuming each strut segment scatters in the same way as an infinite cylinder. The strut scattering from each strut segment is then summed to get the total strut scattering. Struts with circular cross sections were considered in the NEC Reflector Antenna Code. General cross section struts have been included in the updated version of the code. Also, two sources of the incident field are used for the strut scattering. One is the geometrical optics fields from the reflector surface (as used in the NEC Code) and the other one is the direct illumination by the feed.

Capabilities for analyzing Cassegrain and Gregorian dual-reflector antennas are also included in the updated Reflector Antenna Code [9]. A two-step procedure is required to calculate the patterns of such reflectors. The subreflector patterns are calculated first by GTD and then used as the feed for the main reflector. Then the patterns of the main reflector which is a paraboloidal surface are calculated.

In the NEC Reflector Antenna Code, the radiation patterns of the feed horn with circular symmetry can only be calculated by aperture integration. Only the main beam and first few sidelobes are provided accurately by this approach. No back lobe information is given. A body of revolution moment method code has been implemented in the new Reflector Antenna Code which can be used to obtain the complete 360° patterns of horns with body of revolution such as conical horns, conical corrugated horns, dual-mode horns, and circular waveguides.

Tolerance effects for the reflector surface can be deterministically analyzed in the OSU Reflector Antenna Code. Surface errors can be input in the code and the scattered field from the surface errors and the total field of the reflector can be calculated.

With multi-beam systems for satellite communications, array feeds and defocussed feeds are increasing in importance. Reflector antennas with array feeds can also be modeled with the code. A ray tracing technique is applied to obtain the aperture fields of the reflector antenna with an arbitrary feed location. The computer code permits the reflector patterns to be calculated by either the principle of

superposition for each feed element or a two-step procedure. The straight-forward two-step procedure first computes the feed pattern of the complete array and then treats this pattern as that from a single feed to compute the reflector pattern either by AI or by GTD. This procedure is valid provided the array has a well defined phase center at ranges corresponding to the distance from the feed to the surface of the reflector. When appropriate, the two-step procedure is more efficient. Some capabilities of the original NEC Reflector Antenna Code are summarized as follows:

1. A general reflector rim shape may be used (piecewise linear). Irregular or jagged shapes may not work because of complicated shadowing of the spillover fields and complicated limits for the AI.

2. The required input data for the feed pattern is minimized by piecewise linear pattern fitting. The feed may be linearly polarized with any orientation or circularly polarized.

3. A feed pattern option is available for a dominant mode horn feed (either corrugated or smooth) in which the horn dimensions are input.

4. Storage and computation time of aperture data for AI is minimized by using a principal rectangular grid and interpolation of the aperture field.

5. The combined AI/GTD approach gives full pattern capability for both far field and near field data.

6. The efficiency of field computations is maximized by the use of GTD for wide pattern angles and the use of the rotating grid method for far field computations at small angles (main beam region).

7. Feed blockage is simulated by a physical optics model of a rectangular or a circular disk.

8. Scattering from feed struts with circular cross section and piecewise linear axes can be modeled.

9. The capability is provided to directly input a linearly polarized aperture field. No feed pattern is then required.

The GTD and AI approaches used for the code have a basic limitation on the minimum size reflector that can be modeled. This limitation is one the order of 3 to 5 wavelengths for the reflector diameter. By comparison to the exact solution for a circular disk the code has been shown to be reasonably accurate for reflector diameters as small as 3 wavelengths [3]. There is no basic limitation on the maximum size of the reflector for the basic analysis. (In the code, the reflector

antenna must have sufficiently good tolerances, especially at high frequencies, so that it can be accurately modeled by the code.)

Some new capabilities of the updated OSU Reflector Antenna Code are summarized as follows:

1. A multi-point GTD method is used to improve the calculation efficiency on the wide angle sidelobes of the reflector.

2. An extended aperture integration (AIE) is implemented to improve the accuracy of aperture integration for offset reflector antennas.

3. The radiation patterns of Cassegrain and Gregorian dual-reflector antennas can be calculated.

4. Complete patterns of horn antennas with circularly symmetric geometry can be calculated.

5. The patterns of reflector antennas with array feeds or defocussed feeds can be calculated.

6. Feed scattering effects can be calculated for offset reflectors by the Extended Aperture Integration method.

7. Struts with general cross section can be modeled.

8. Direct feed incidence on the struts is included.

9. Surface tolerance effects on the reflector patterns can be modeled.

The practical limitations on this version of the code can be summarized as follows:

1. The grid size used for aperture integration must be chosen sufficiently small to give a good representation of the aperture field distribution.

2. Array variables associated with the rim data, the principal grid, the feed pattern and the output pattern must be given sufficient dimensions for the required input data. Similar restrictions apply to the dimensions of struts and plates.

3. The strut diameters should be no more than 10 wavelengths.

The use of the OSU Reflector Antenna Code is described in this report. Examples are included at appropriate places so that the user can learn how to run the code.

1-5

## 2. OUTPUT FROM THE CODE

For far field calculations or for near field calculations with constant range (LRANG=true), the total field is converted to principal and cross polarized components as referred to the polarization of the field components from a Huygen's source, definition 3 of Ludwig [10]. For near field calculations with constant-Z plane (LRANG=false), the field is expressed in rectangular components.

### 2.1. Magnitude and Phase Outputs

Far field calculations can be made with or without the range factor and this is controlled by the input logical variable LRFCT. If the range factor is suppressed (LRFCT=false) the magnitude and phase outputs express the antenna field pattern. For far field calculations including the range factor (LRFCT=true) or for near field calculations, the magnitude and phase outputs are expressed as the electric field relative to the aperture field level.

### 2.2. dB Output

The dB output of the code is expressed as antenna directive gain relative to isotropic when feed patterns are input with the FD:Command. When aperture fields are directly input by the AP:Command, the dB output simply gives the relative electric field levels.

In order to determine the antenna gain levels the radiated power is determined by integrating the power density radiated by the feed. Thus

$$P_{rad} = \iint \frac{|E_f|^2}{Z_0} R^2 \sin\theta \; d\theta \; d\phi \tag{1}$$

where the field of the feed is given by

$$E_f = \frac{F}{R} g_f(\psi,\phi) e^{-jkR} \tag{2}$$

and F is the focal length of the reflector. This gives

$$P_{rad} = \frac{F^2}{Z_0} P_{RAD} \tag{3}$$

where $Z_0$ = 376.7 ohms and

$$P_{RAD} = \iint |g_f|^2 \sin\theta \; d\theta \; d\phi \tag{4}$$

is calculated in the code. Consequently, the absolute field level is given by

$$|E| = |E_0| \left(\frac{P_t}{P_{rad}}\right)^{1/2}$$  (5)

where $E_0$ is the magnitude output of the code (relative electric field) and $P_t$ is the actual radiated transmitter power (in Watts). The actual power density (in Watts per square meter) is given by

$$S_p = \frac{|E|^2}{Z_0} = \frac{|E_0|^2}{Z_0} \frac{P_t}{P_{rad}} = \frac{|E_0|^2 P_t}{F^2 P_{RAD}}$$  (6)

The antenna gain G can be determined from the power density as follows:

$$S_p = \frac{P_t G}{4\pi R^2}$$  (7)

where R is the range as referred to the reference point $X_{ref}$. In the code the reference point is chosen on the aperture plane, thus

$$X_{ref} = (X_0, Y_0, ZOP)$$

where $X_0$ and $Y_0$ are at the center of the aperture,

$$ZOP = \frac{(R_{max})^2}{4F}$$

and $R_{max}$ is the maximum radius to the reflector rim. The antenna gain can be determined from the magnitude output of the code as follows:

$$G = \frac{4\pi R^2 |E_0|^2}{F^2 P_{RAD}}$$  (8)

The information for F and $P_{RAD}$ are included in the variable

$$REFDB = 10 \log \frac{4\pi \lambda^2}{F^2 P_{RAD}}$$

which is used in the DBPHS subroutine to calculate far field gain and is given as output from the code. For far field calculations including the range factor (LRFCT=true) or for near field calculations, the near field gain is calculated from Equation (8) by using the DBPHS subroutne with

$$REFNF = 10 \log \frac{4\pi \ \lambda^2}{F^2 P_{RAD}}$$

For radiation hazard applications the HZ: Command is used to calculate the power density in milliwatts per square centimeter, total electric field in Volts per meter and near field gain in dB. The power density is based on the electric field and free space impedance.

## 3. APPLICATIONS OF THE CODE

The reflector code can be used for the following basic applications:

1. Pattern prediction of existing reflector antennas.

2. Reflector antenna design.

3. Radiation hazard calculations.

4. EMC or coupling calculations with small antennas.

The far field capability of the code is used for applications 1 and 2 listed above. For pattern prediction it is necessary to have sufficient information about the reflector dimensions and feed pattern. For antenna design the code can be used in an iterative manner to seek a practical design having a given pattern performance goal. Or, the code can be used to give a more accurate prediction of the performance of a design obtained from more approximate techniques.

The near field capability of the code can be used for EMC and radiation hazard applications. The code can accurately calculate the field at virtually any point that is at least one diameter from the reflector. Since the code is efficient even for near field computations it eliminates the necessity to rely on approximate techniques as has usually been done in the past.

For radiation hazard applications the code is used to calculate the level of the electric field and the power density at the near field point as discussed in Section II.

For EMC or coupling calculations, the coupling can be calculated by using the CL: Command. The output of the code will then give the ratio of the received power to transmitted power between the reflector and the linear antenna in dB. The coupling output data which result from use of the CL: Command are calculated from the reaction integral involving the fields of the reflector and the currents of the linear antenna.

## 4. COMMAND WORD SYSTEM

The method used to input data into the computer code is based on a command word system. With this system the code stores the previous input data such that one need only input that data which needs to be changed from the previous execution. This is especially convenient when more than one problem is to be analyzed during a computer run. Also, there is a default list of data built into the NX: Command which is automatically executed before any other command. Thus the code will run the default case by simply executing the XQ: Command.

It is recommended to not execute any command for which input data is not needed. For example, the NF: Command usually is not needed for far field data.

All angles in these commands are input in degrees. For distance or length variables, the units of length are specified according to the value of IUNIT in the DG: Command, unless otherwise noted. Consequently, the DG: Command should be executed before any other command in which the input units need to be specified. However, the CM: or CE: Command can be executed before the DG: Command because it does not require input units. If the DG: Command is not used, the unit of length is inches.

NOTE: THE DG: COMMAND SHOULD NORMALLY BE THE FIRST COMMAND (AFTER CM: OR CE:) IN A DATA FILE!

The following sections define in detail each command word and the variables associated with them. A block diagram is given for each command word which shows the way to input the data associated with that command.

```
----------------------------------------------------------
Command CM:  or CE:   COMMENTS ON OUTPUT DATA
----------------------------------------------------------
```

This command enables the user to write comments along with the output data of the code.  If the CM:  command is input, comment cards can then be read and the corresponding comments will be written as part of the code output.  Each comment card except the last in a sequence must have 'CM:'  for its first three characters.  The last comment card in the sequence must have 'CE:'  for its first three characters:  then the code returns for another possible command word.

1.  READ:  IR(I),I=1,24

   a)  IR(I):  This is a dimensioned array of up to 72 typed characters (assuming 3 characters per word) which compose the desired comment.  As stated before, the first three characters must be 'CM:'  for all comment cards except 'CE:'  for the last comment card in the sequence.

```
------------------------------------------------------------
Command NX:   DEFAULT INPUT DATA
------------------------------------------------------------
```

This command resets the input data of the code to that of the default
case.  Thus the user can read this command word followed by the XQ:
Command to run the default data at any time.  Also, this command is
automatically executed before any other command.  Thus the default case
can be run as the first case by a single XQ:  Command word.


BLOCK DIAGRAM FOR DEFAULT DATA

```
          ┌─────────────────────┐
          │  Command NX:        │
          └──────────┬──────────┘
                     │
          ┌──────────┴──────────┐
          │Reset Default Data   │
          └─────────────────────┘
```
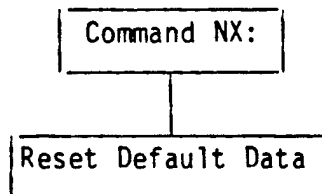
---------------------------------------------------------
Command XQ:  EXECUTE CODE
---------------------------------------------------------

This command is used to execute the reflector code so that the
fields of the reflector may be computed and output.  After execution the
code returns for another possible command word.

```
            ┌─────────────────┐
            │ Command XQ:     │
            └────────┬────────┘
                     │
       ┌─────────────┴──────────┐
       │ Execute Computer Code  │
       └────────────────────────┘
```
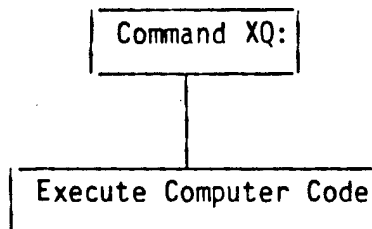
```
-------------------------------------------------------------
Command EN:  STOP CODE
-------------------------------------------------------------
```

This command stops execution of the code.

```
              ┌─────────────────┐
              │  Command EN:    │
              └────────┬────────┘
                       │
                       │
              ┌────────┴────────┐
              │Stop Computer Code│
              └─────────────────┘
```
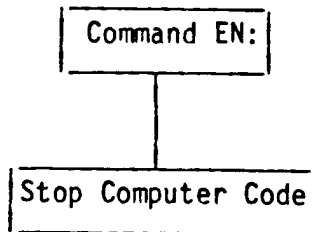
```
-----------------------------------------
Command DG:  REFLECTOR (DISH) GEOMETRY
-----------------------------------------
```

This command enables the user to specify the reflector type and its geometry, that is, the shape and dimensions of the reflector. The rectangular grid used for aperture integration or physical optics integration is also specified by this command.

All units are specified according to the value of IUNIT. Since all units are specified according to the value of IUNIT in this command, this command should be executed before any other command in which the input units need to be specified. The CM: or CE: command can be executed before the DG: command because it does not require input units.

NOTE:  THE DG:  COMMAND SHOULD NORMALLY BE THE FIRST COMMAND IN A DATA
       FILE!

1. READ:  NTYPE

   a) NTYPE:  This integer variable specifies the type of
      reflector surface as follows

                    1:  parabolic reflector surface
      NTYPE =      10:  hyperbolic reflector surface
                   11:  elliptic reflector surface

      For NTYPE > 10, LAIC (in TO:  Command) must be
      false.

2. READ:  IUNIT,F,GRIDX,GRIDY,D,NRIM

   a) IUNIT:  This integer variable indicates the units
      for the input data as follows:

                    1-> meters
      IUNIT = <     2-> feet
                    3-> inches

   b) F:  This real variable defines the focal distance of
      the reflector as shown in Figure 1.

c) GRIDX,GRIDY: These real variables define the rectangular grid dimensions, GRIDX and GRIDY, as shown in Figure 2b. The rectangular grid is used for aperture integration or surface current integration (physical optics) and thus its size must be sufficiently small to provide a reasonable representation of the aperture field or surface current. However, the grid dimensions may be large in wavelengths for aperture integration. The grid dimensions GRIDX and GRIDY together with the reflector size (aperture size for aperture integration) control the number of grid lines MMAX, NMAX used for integration. The maximum number of grid lines is limited by MDGRID. At least 3 grid lines must be used in the code. Presently, 3 < MMAX < 200, 3 < NMAX < 200.

Note that more grid lines are required when the rotating grid is used for off-principal plane cuts. Approximately 50% more grid lines are required for PHI-cuts near 45° and odd multiples of 45°.

d) D: This real variable defines the diameter of the reflector. If the diameter is read as a positive value (D>0), the reflector is assumed to be circular as shown in Figure 1 and the code generates the rim points. If the diameter is zero, a general rim shape may be read. If the diameter is -1, an elliptic rim is assumed and the lengths of the major and minor axes of the ellipse are read.

e) NRIM: This integer variable defines the number of input rim points. For circular rims (D>0), read NRIM=0 for automatic calculation of NRIM in the code. Presently 3<NRIM<627.

3. READ: FC2

This statement is skipped if NTYPE < 10

a) FC2: This real variable specifies the distance between the focii of hyperbolic and elliptic reflectors as shown in Figure 1.

4.  READ:  DAA,DBB

    This statement is used for D=-1 only (elliptic rims).
    The geometry of the elliptic rim is shown in Figure 3.

    a) DAA:  This real variable defines the length of the major axis of
       the elliptic rim.

    b) DBB:  This real variable defines the length of the minor axis of
       the elliptic rim.


5.  READ:  ((RIM(NE,N),N=1,2),NE=1,NRIM)

    This statement is used for D=0 only (general rim shape).

    a) RIM(NE,N):  This doubly dimensioned real variable is used to
       specify the location of the NE-th corner of the projected
       piecewise linear aperture rim as shown in Figure 2a.  It is input
       on a single line with the real numbers being the x, y coordinates
       of the corner which correspond to N=1,2, respectively, in the
       array.

    Note:  Rim points must be input in the counterclockwise sense.

           A smooth portion of the reflector rim can be simulated by
           using a sufficient number of rim points.  A recommended
           criterion for the length (RIML in wavelengths) between two
           consecutive rim points on the smooth rim portion is as
           follows:

    $$\frac{RIML}{\lambda} = \min \left( \sqrt{\frac{F}{2\lambda}} \, , \, 0.1F/\lambda \right)$$

           where F is the focal distance in wavelengths in read
           statement 2.

BLOCK DIAGRAM FOR REFLECTOR GEOMETRY

```
                        ┌──────────────────┐
                        │ Command  DG:     │
                        └────────┬─────────┘
                                 │
                        ┌────────┴─────────┐
                        │ Read:  NTYPE     │
                        └────────┬─────────┘
                                 │
            ┌────────────────────┴──────────────────────┐
            │ Read:  IUNIT,F,GRIDX,GRIDY,D,NRIM          │
            └────────────────────┬──────────────────────┘
                                 │
                               ╱   ╲
                             ╱       ╲
                           ╱           ╲
                         ╱               ╲            FALSE
                        ⟨  IF (NTYPE>10)   ⟩──────────────────┐
                         ╲               ╱                    │
                           ╲           ╱                      │
                             ╲       ╱                        │
                               ╲   ╱                          │
                                │ TRUE                        │
                        ┌───────┴──────┐                      │
                        │ Read:  FC2   │                      │
                        └───────┬──────┘                      │
                                │◄────────────────────────────┘
                                │
                              ╱   ╲
         D = -1             ╱   *   ╲            D > 0
     ┌─────────────────────⟨  D = ?   ⟩─────────────────────┐
     │                      ╲         ╱                      │
     │                        ╲     ╱                        │
     │                          ╲ ╱                          │
     │                           │                           │
┌────┴──────────────────┐       │ D = 0      ┌──────────────┴──────────┐
│ Read:  DAA,DBB         │       │            │ Calculate RIM(NE,N)     │
│ for elliptic rim       │       │            │ for circular rim        │
└────────────────────────┘       │            └─────────────────────────┘
                    ┌────────────┴────────────────────────────┐
                    │ Read:  ((RIM(NE,N),N=1,2),NE=1,NRIM)     │
                    │        for general rim shape             │
                    └─────────────────────────────────────────┘
```
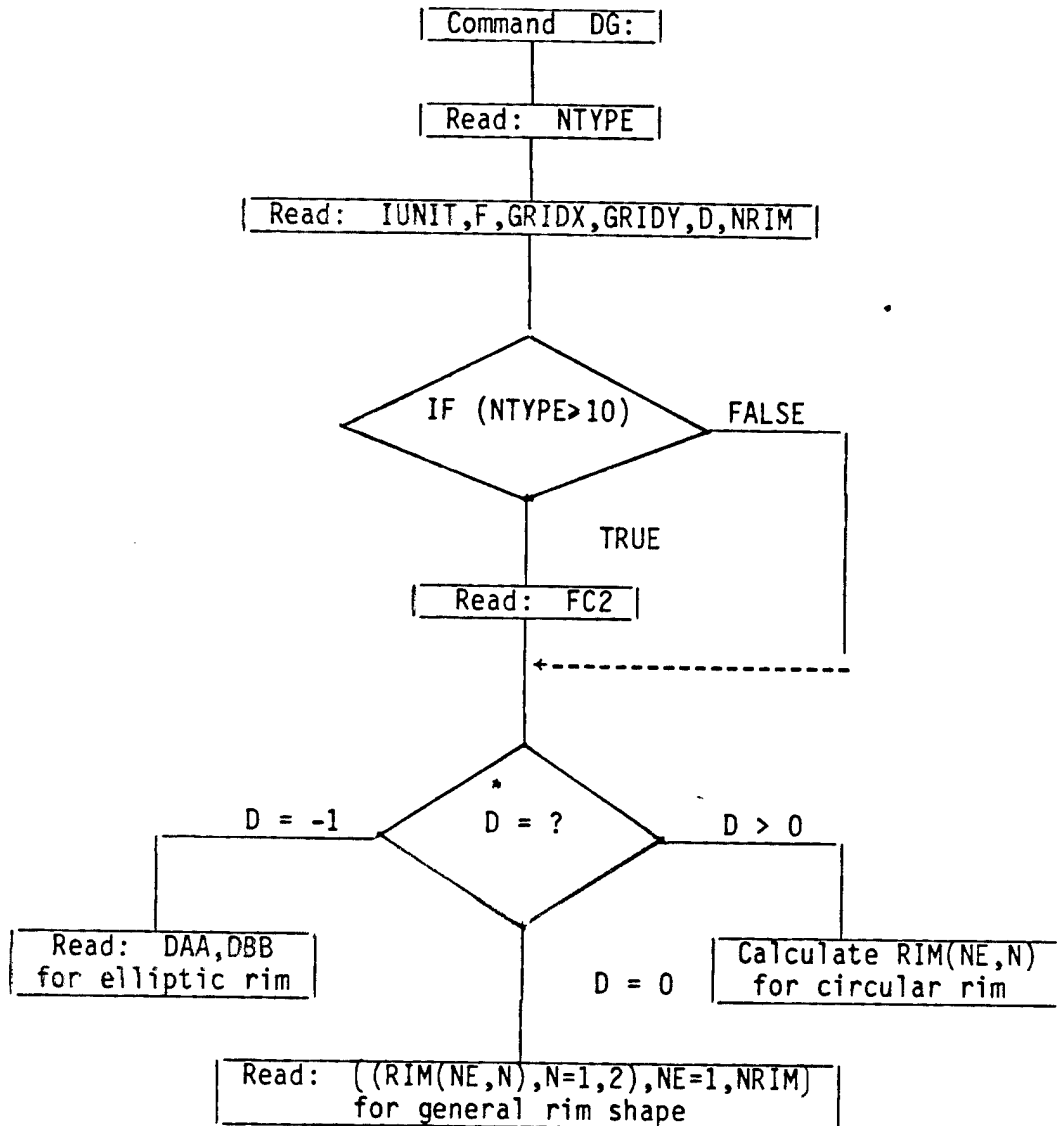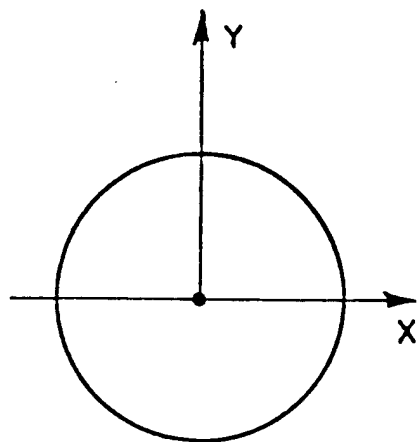
(a) Front view
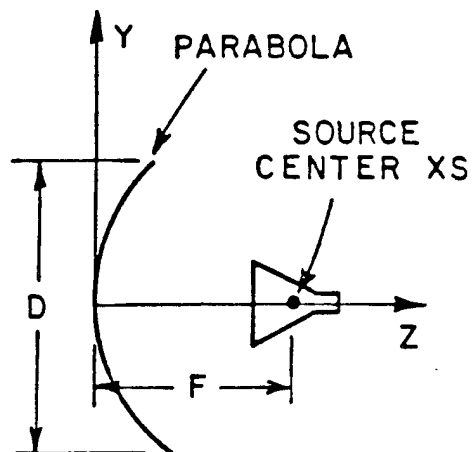
(b) Side view of parabolic reflector
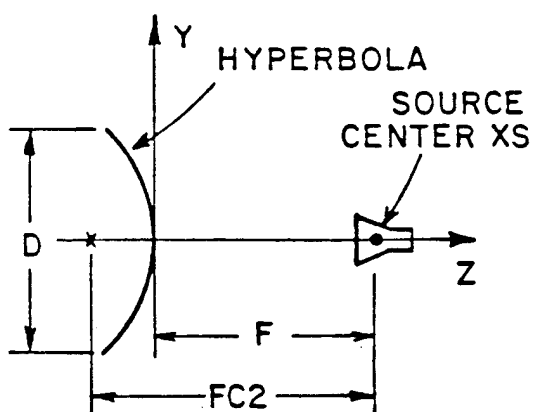
(c) Side view of hyperbolic reflector
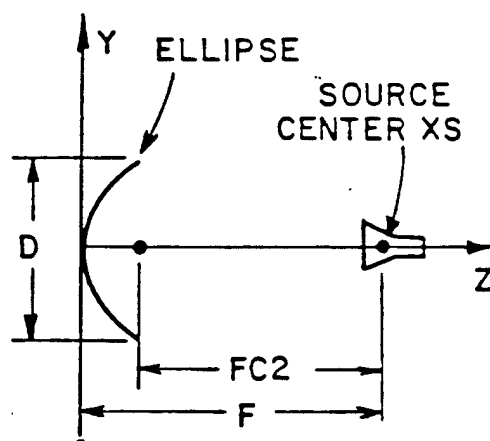
(d) Side view of elliptic reflector

Figure 1. Center-fed circular reflector. (D>0)
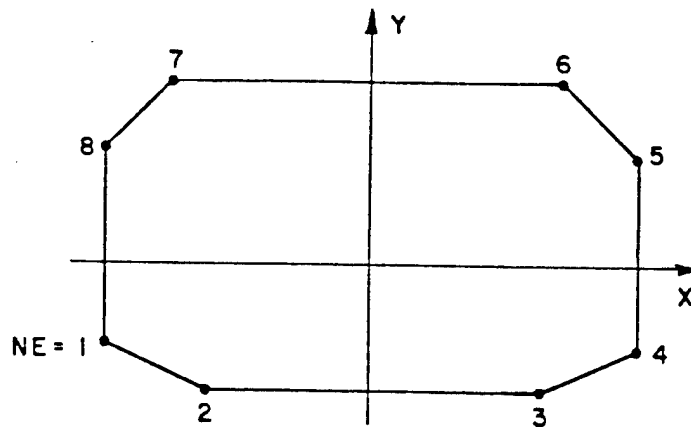
RIM $(NE, 1) = X_{NE}$

RIM $(NE, 2) = Y_{NE}$

PROJECTED RIM OF REFLECTOR

(a) Non-circular: D=0
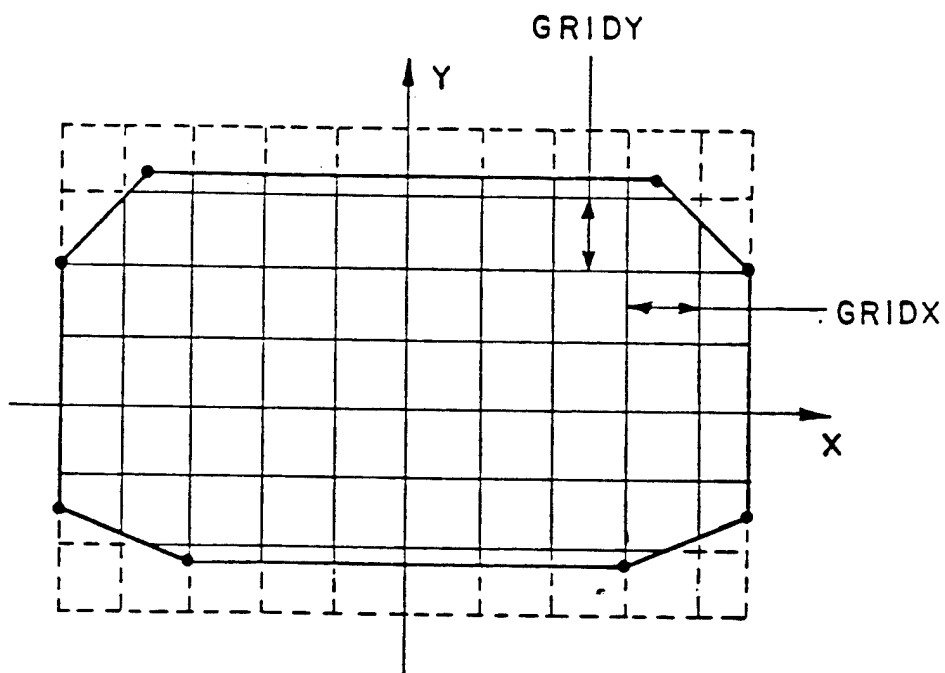


(b) Principal grid

Figure 2. Reflector rim geometry and principal rectangular grid.
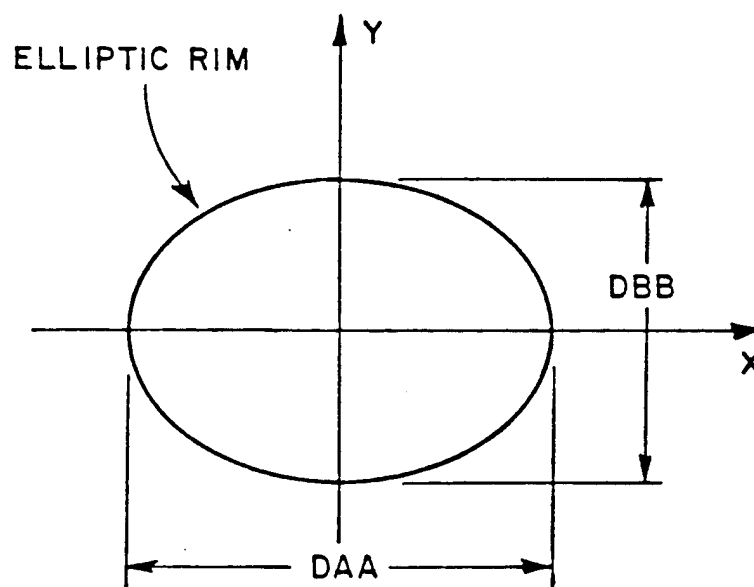
Figure 3. Front view of elliptic rim reflector (D=-1).

Examples of a 24.0" parabolic reflector antenna are given in Appendix B to illustrate how the DG: Command in the Reflector Antenna Code is used to obtain reflector antenna patterns. Some often used commands other than the DG: Command are also used in those examples. The first example of this section shows how to obtain the radiation patterns of Cassegrain reflector antennas. Similar processes can be used to calculate patterns of Gregorian reflector antennas. When running the subreflector patterns, one has to set the phase center X00 at the second focus of the subreflector in order to get constant phase in the G.O. region. The phase center X00 is input from the NF: Command and is shown in Figure 4.
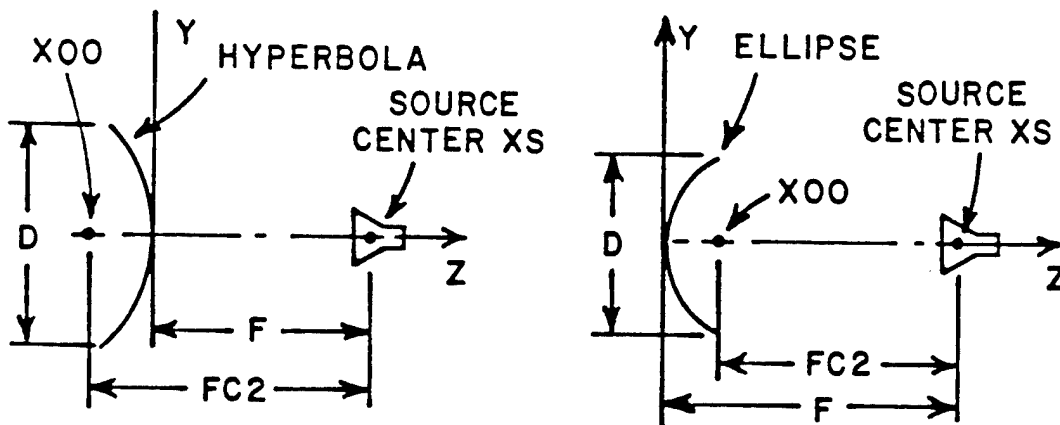


Figure 4. Phase center for subreflector pattern calculation.

The second example shows how to input a reflector with elliptic rim shape by setting D=-1.

Example 1:

This example illustrates how to calculate the patterns of a Cassegrain reflector antenna by a two-step procedure from the reflector antenna code. The geometry of the antenna is given in Figure 5 with

$$D_H = 2.919" \quad , \quad D_M = 24.0"$$
$$F = 4.163" \quad \quad F_C = 4.728"$$
$$F_M = 8.0" \quad \quad f = 38.0 \text{ GHz.}$$

The primary feed is a circular TE11 mode conical horn with aperture diameter 1.2", flare angle 14.9° and phase center 0.565" behind the aperture. The primary feed is located at the real focal point of the reflector antenna system. The patterns of this conical horn are calculated by the AP: Command and the results are given in the example of the section of the AP: Command.

The subreflector of the Cassegrain antenna is a hyperbolic reflector (NTYPE=10) and its patterns are calculated first. The input data of this run are given in Table 1. Note that the feed data in the FD: Command are obtained from the example in the AP: Command section by setting LFDOUT=T in the PZ: Command. The computed and measured principal patterns are given in Figure 6. The computed subreflector pattern data are stored in Unit #7 by LFDOUT=T in PZ: command for the purpose of the next run.

After the subreflector patterns are obtained, the patterns of the parabolic main reflector (NTYPE=1) can be calculated by using the subreflector patterns as the feed patterns. The input data of this run are given in Table 2. The data in FD: command are obtained from previous run. The calculated and measured patterns of the main reflector are given in Figure 7.
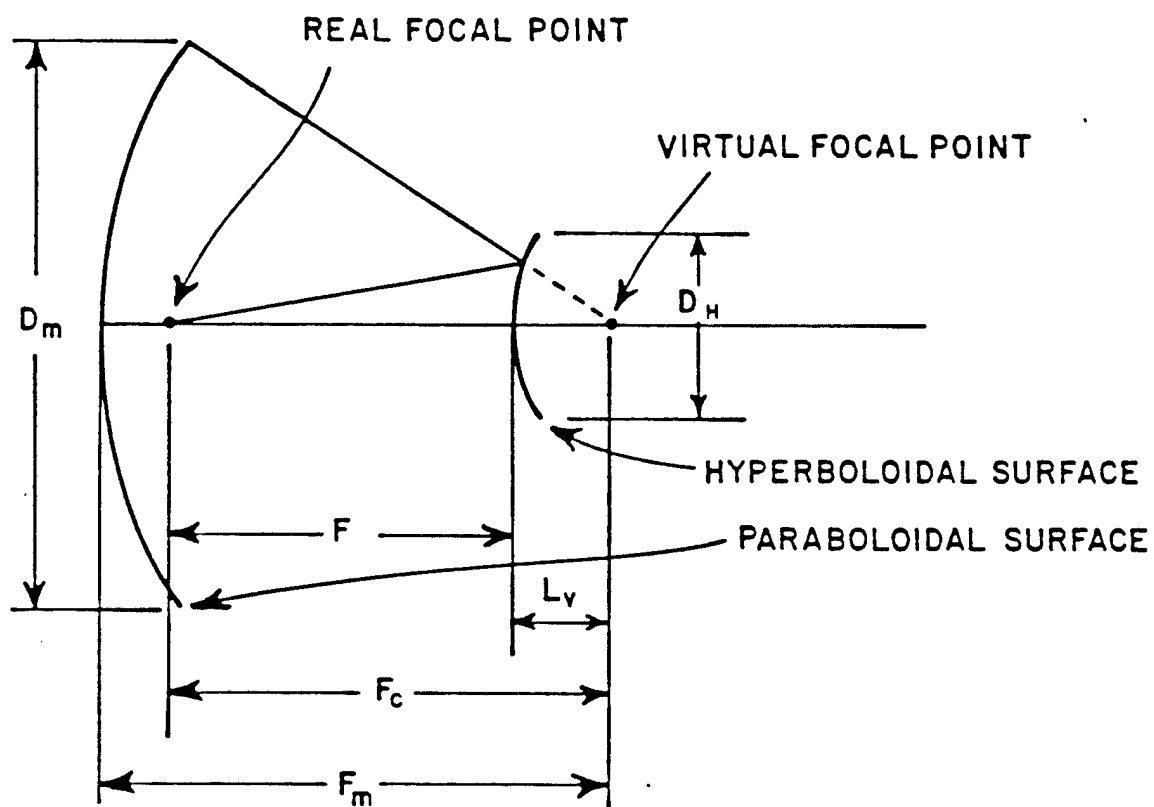
Figure 5. Geometry of Cassegrain reflector antenna.
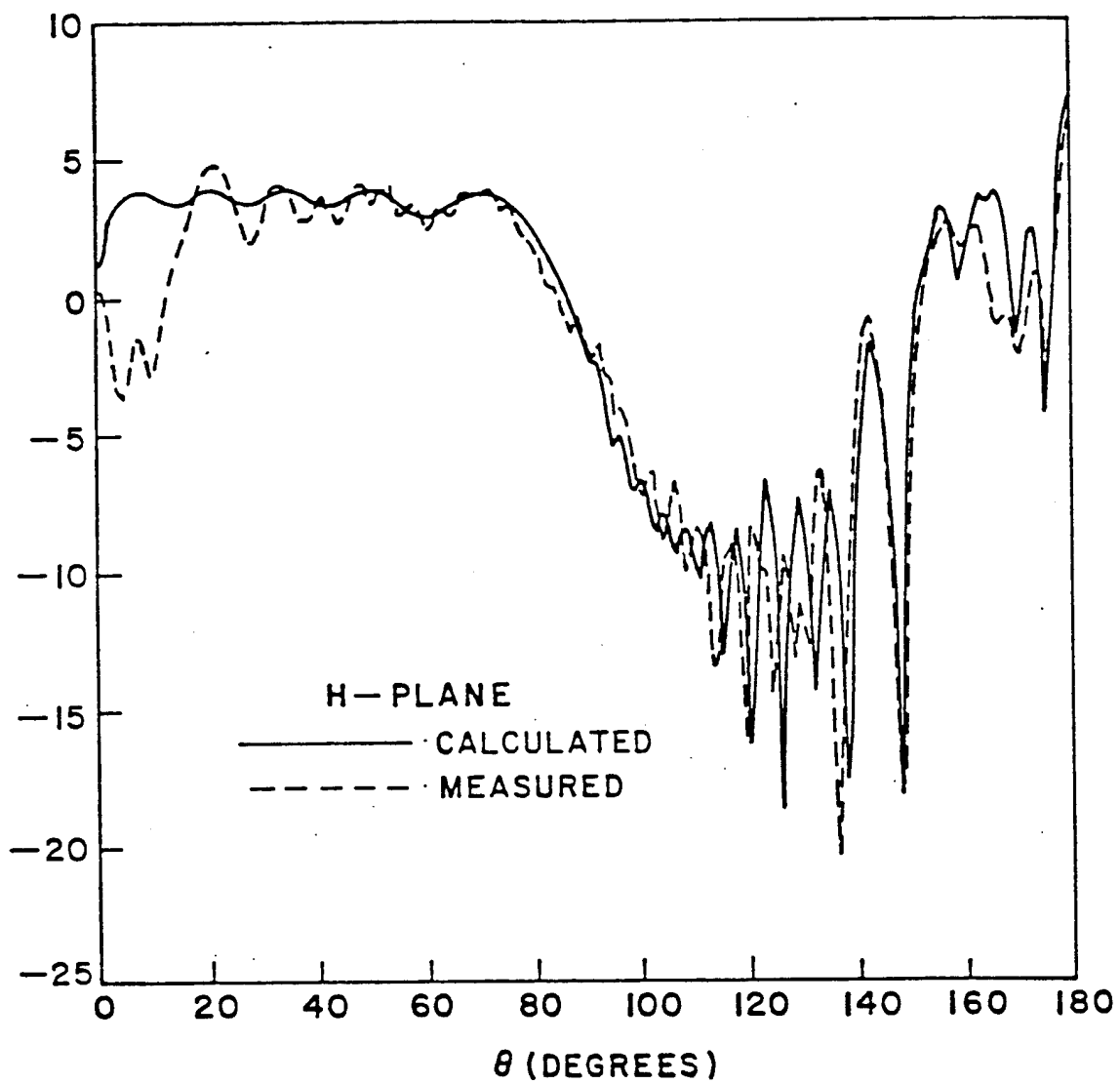
Figure 6. H-plane and E-plane patterns of hyperbolic subreflector of the example.
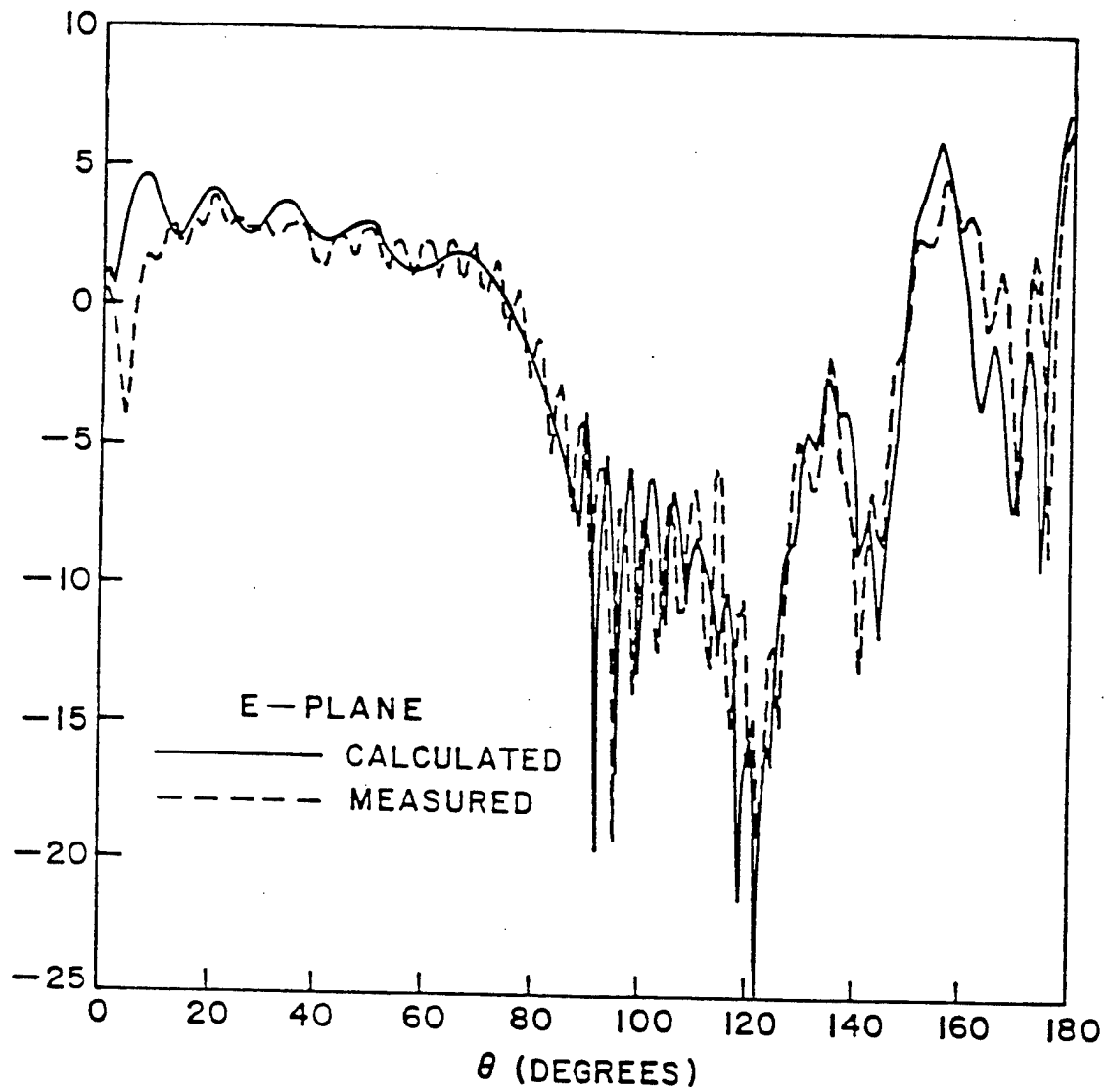
Figure 6.   (Continued.)
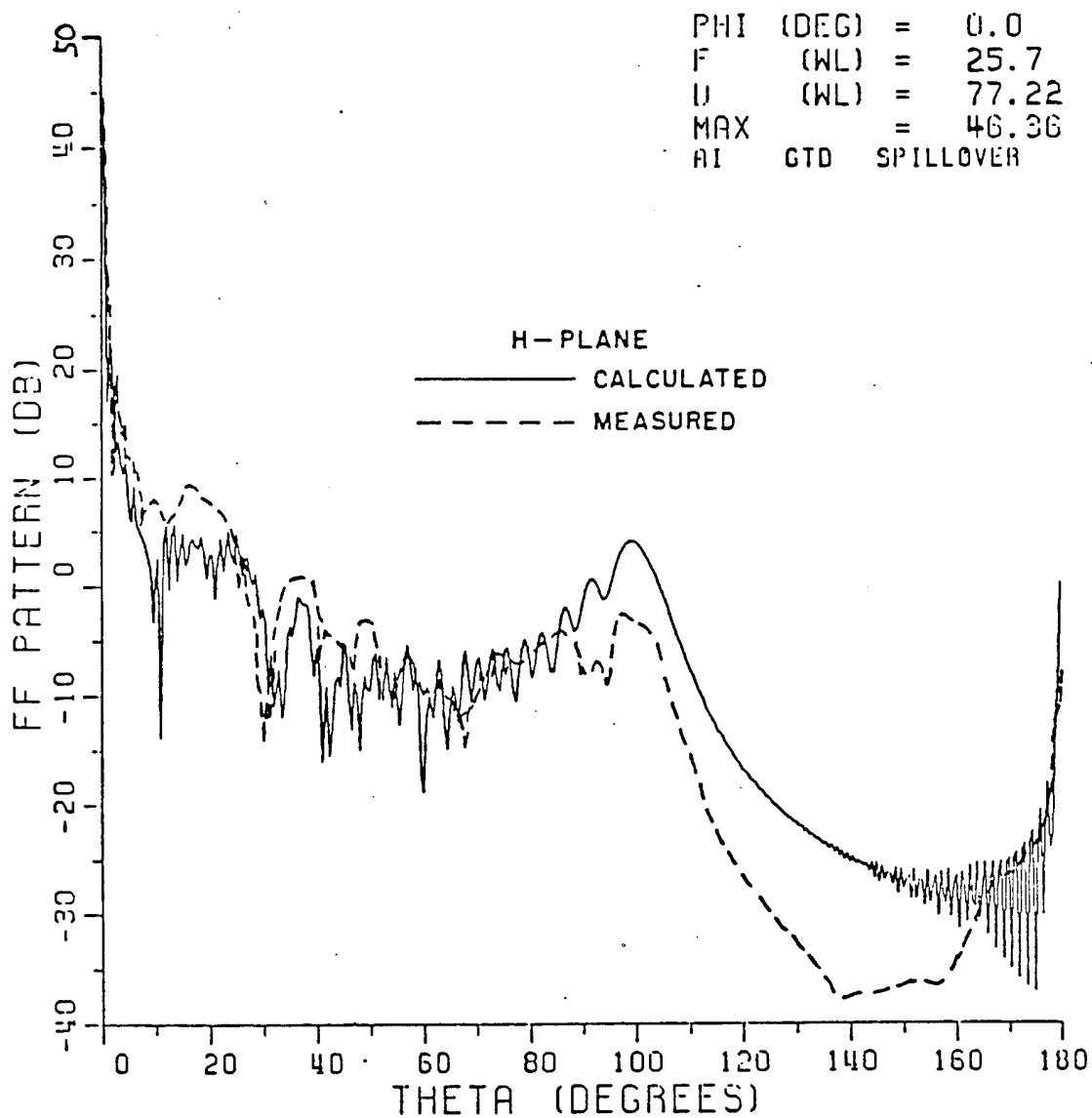
Figure 7. H-plane and E-plane patterns of main reflector of the example.

PHI (DEG) = 90.0
F (WL) = 25.7
D (WL) = 77.22
MAX = 46.37
III GTD SPILLOVER
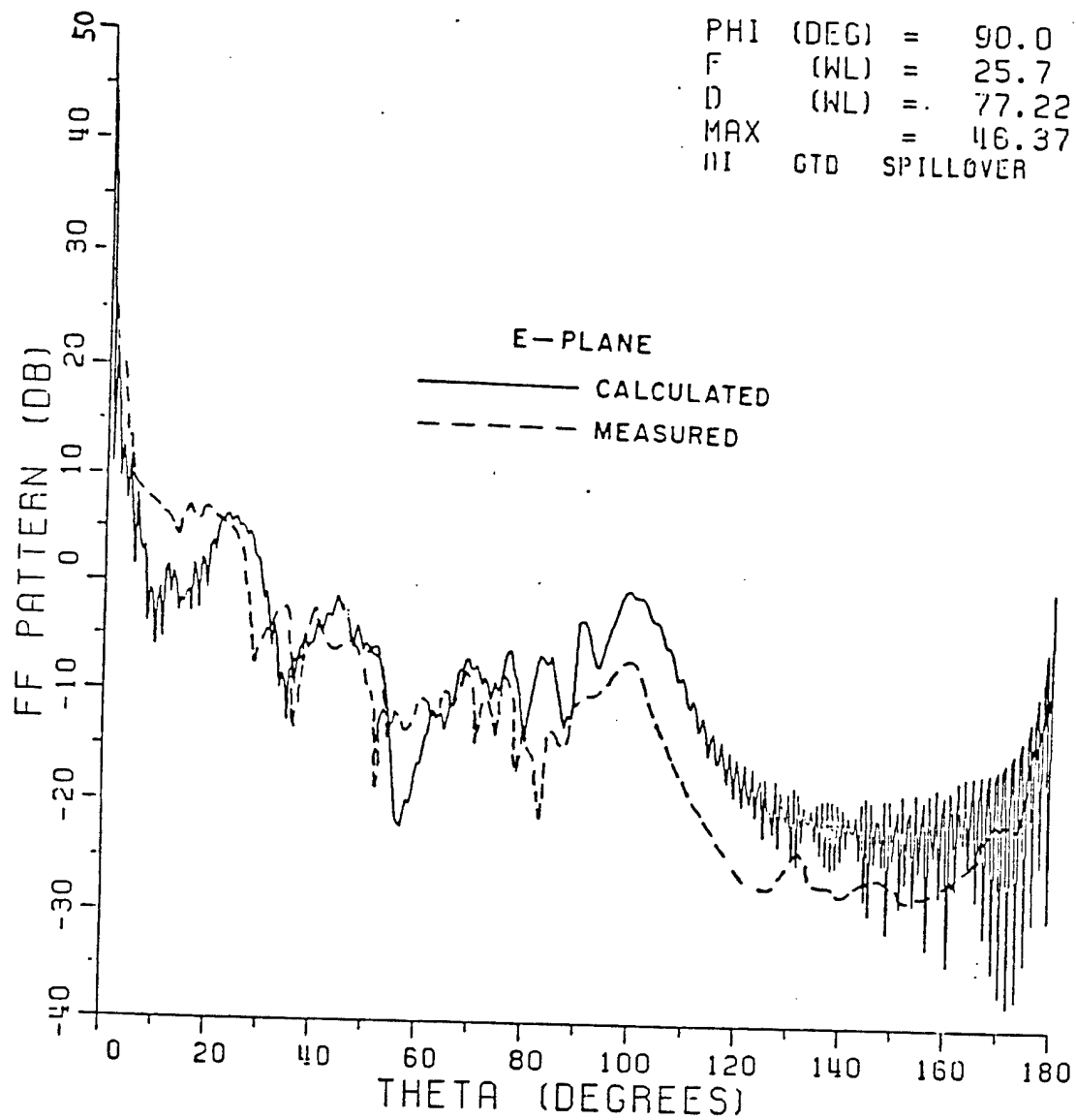
Figure 7. (Continued.)

## TABLE 1

## INPUT DATA FOR PATTERN CALCULATION OF SUBREFLECTOR

```
CM:          *** SUBREF.DAT ***
CM:          CONICAL HORN FEED WITH
CM:          APERTURE DIAMETER=1.2"
CM:          FLARE ANGLE= 14.9 DEG.
CM:  FEED DATA CALCULATED BY AP: IN CONI.DAT
CE:       SUBREFLECTOR PATTERN CALCULATION
DG:
10
3 - 4.163  0.5  0.5  2.919  0
4.728
FQ:
1 38.0
FD:  AD = 0.275"
0    T
T    0    T    1    90.    0.275    F
3    0.   45.   90.
91
       0.000       35.510       71.481      -47.871     -110.173
       1.000       35.478       71.505      -47.992     -111.291
       2.000       35.383       71.578      -48.288     -112.285
       3.000       35.223       71.701      -48.770     -113.173
       4.000       34.999       71.875      -49.454     -113.969
          .            .            .            .            .
          .            .            .            .            .
          .            .            .            .            .


              ( see the example of AP: Command )


          .            .            .            .            .
          .            .            .            .            .
          .            .            .            .            .

      86.000        1.515     -132.115      -70.116      131.990
      87.000        1.319     -132.008      -70.201      132.123
      88.000        1.134     -131.931      -70.306      132.218
      89.000        0.963     -131.885      -70.430      132.274
      90.000     -300.000        0.000     -300.000        0.000
NF:
0. 0. -0.565           ! PHASE REFERENCE OF SUBREFLECTOR PATTERNS
F
F
TO: LAIC=F,LAIS=F,LPO=F,LFEED=T,LGTD=T,LREFL=T
F    90.    1.
F    32    32    1    48
F    F    F    F    0
F    T    F    0.8
F    F    F    T    T    T    0.    0.
T    T    0.    0.
PZ:  LFDOUT=T
3
0. 45. 90.
0. 180. 1.
T
PP:
1
1  1
1  2
XQ:
```

# TABLE 2

## INPUT DATA FOR PATTERN CALCULATION OF MAIN REFLECTOR

```
CM:          *** MAIN.DAT ***
CM:  CALCULATE MAIN REFLECTOR PATTERNS
CM: FEED DATA CALCULATED FROM SUBREF.DAT
CE:          SUBREFLECTOR BLOCKAGE
DG:
1
3   8.0   0.5   0.5   24.   0
FQ:
1 38.0
FD:
0    T
T :  0    T   1   90.   0.   F
3   0.   45.   90.
181
```

|        |       |        |          |          |
|--------|-------|--------|----------|----------|
| 0.000  | 0.967 | 8.395  | -123.560 | -71.142  |
| 1.000  | 1.082 | 7.778  | -90.530  | -46.993  |
| 2.000  | 1.435 | 6.196  | -105.978 | -64.826  |
| 3.000  | 1.956 | 4.145  | -124.967 | -161.925 |
| 4.000  | 2.550 | 2.168  | -104.320 | -176.473 |
| 5.000  | 3.121 | 0.639  | -127.370 | -17.627  |
| 6.000  | 3.593 | -0.267 | -125.602 | -66.099  |
| 7.000  | 3.874 | 1.039  | -131.047 | -101.178 |
| 8.000  | 3.889 | 1.309  | -131.781 | 32.642   |
| 9.000  | 3.826 | 1.901  | -123.943 | 47.900   |
| 10.000 | 3.706 | 2.646  | -121.349 | 50.985   |
| 11.000 | 3.559 | 3.342  | -121.309 | 51.826   |
| 12.000 | 3.417 | 3.801  | -123.247 | 51.218   |
| 13.000 | 3.311 | 3.897  | -128.319 | 46.439   |
| 14.000 | 3.265 | 3.610  | -141.442 | -21.585  |
| 15.000 | 3.285 | 3.030  | -129.788 | -104.446 |
| 16.000 | 3.364 | 2.336  | -125.346 | -109.515 |
| 17.000 | 3.486 | 1.728  | -124.394 | -109.249 |
| 18.000 | 3.619 | 1.362  | -125.649 | -106.278 |
| 19.000 | 3.734 | 1.312  | -129.968 | -96.757  |
| 20.000 | 3.807 | 1.571  | -138.812 | -35.522  |
| 21.000 | 3.825 | 2.058  | -130.777 | 41.253   |
| 22.000 | 3.785 | 2.646  | -126.023 | 52.585   |
| 23.000 | 3.692 | 3.184  | -124.197 | 56.532   |
| 24.000 | 3.571 | 3.539  | -124.273 | 58.040   |
| 25.000 | 3.450 | 3.611  | -126.224 | 57.473   |
| 26.000 | 3.353 | 3.363  | -130.966 | 51.498   |
| 27.000 | 3.303 | 2.844  | -141.060 | -6.898   |
| 28.000 | 3.313 | 2.178  | -132.596 | -86.091  |
| 29.000 | 3.380 | 1.528  | -127.823 | -93.677  |
| 30.000 | 3.481 | 1.041  | -126.098 | -93.787  |
| 31.000 | 3.602 | 0.827  | -126.349 | -90.846  |
| 32.000 | 3.717 | 0.920  | -128.378 | -84.060  |
| 33.000 | 3.804 | 1.285  | -132.875 | -65.842  |
| 34.000 | 3.848 | 1.840  | -136.967 | 0.533    |
| 35.000 | 3.842 | 2.470  | -131.361 | 49.657   |
| 36.000 | 3.785 | 3.049  | -127.601 | 63.333   |
| 37.000 | 3.683 | 3.458  | -125.944 | 69.386   |
| 38.000 | 3.561 | 3.609  | -125.737 | 72.702   |
| 39.000 | 3.437 | 3.449  | -126.908 | 74.274   |
| 40.000 | 3.336 | 2.986  | -129.847 | 72.987   |
| 41.000 | 3.275 | 2.290  | -136.095 | 61.282   |
| 42.000 | 3.266 | 1.480  | -141.183 | -34.845  |
| 43.000 | 3.303 | 0.695  | -132.360 | -69.240  |
| 44.000 | 3.384 | 0.076  | -128.360 | -72.981  |
| 45.000 | 3.494 | -0.283 | -126.469 | -72.019  |
| 46.000 | 3.614 | -0.334 | -125.858 | -69.028  |
| 47.000 | 3.725 | -0.092 | -126.295 | -64.255  |

(TABLE 2 - CONTINUED)

| | | | |
|---|---|---|---|
| 48.000 | 3.812 | 0.403 | -127.713 | -56.796 |
| 49.000 | 3.859 | 1.071 | -130.245 | -43.746 |
| 50.000 | 3.858 | 1.823 | -133.655 | -16.565 |
| 51.000 | 3.815 | 2.567 | -134.714 | 31.628 |
| 52.000 | 3.733 | 3.198 | -128.632 | 34.120 |
| 53.000 | 3.619 | 3.654 | -129.169 | 49.483 |
| 54.000 | 3.483 | 3.859 | -129.912 | 65.635 |
| 55.000 | 3.333 | 3.773 | -130.870 | 83.004 |
| 56.000 | 3.183 | 3.386 | -131.924 | 102.447 |
| 57.000 | 3.054 | 2.719 | -132.968 | 124.615 |
| 58.000 | 2.960 | 1.831 | -133.746 | 149.794 |
| 59.000 | 2.910 | 0.807 | -133.946 | 176.851 |
| 60.000 | 2.909 | -0.248 | -133.464 | -156.837 |
| 61.000 | 2.944 | -1.234 | -132.502 | -133.476 |
| 62.000 | 3.016 | -2.064 | -131.367 | -113.527 |
| 63.000 | 3.117 | -2.672 | -130.256 | -96.411 |
| 64.000 | 3.237 | -3.021 | -129.255 | -81.386 |
| 65.000 | 3.364 | -3.104 | -128.414 | -67.817 |
| 66.000 | 3.480 | -2.933 | -127.677 | -55.402 |
| 67.000 | 3.576 | -2.536 | -127.063 | -43.794 |
| 68.000 | 3.652 | -1.944 | -126.558 | -32.776 |
| 69.000 | 3.704 | -1.196 | -126.146 | -22.209 |
| 70.000 | 3.729 | -0.330 | -125.814 | -11.989 |
| 71.000 | 3.722 | 0.616 | -125.551 | -2.038 |
| 72.000 | 3.674 | 1.612 | -125.368 | 7.701 |
| 73.000 | 3.597 | 2.623 | -125.205 | 17.288 |
| 74.000 | 3.491 | 3.619 | -125.076 | 26.756 |
| 75.000 | 3.357 | 4.575 | -124.969 | 36.131 |
| 76.000 | 3.189 | 5.479 | -124.873 | 45.436 |
| 77.000 | 2.986 | 6.314 | -124.777 | 54.678 |
| 78.000 | 2.759 | 7.052 | -124.671 | 63.841 |
| 79.000 | 2.509 | 7.679 | -124.541 | 72.907 |
| 80.000 | 2.238 | 8.180 | -124.384 | 81.844 |
| 81.000 | 1.927 | 8.584 | -124.189 | 90.634 |
| 82.000 | 1.596 | 8.842 | -123.951 | 99.207 |
| 83.000 | 1.247 | 8.946 | -123.668 | 107.512 |
| 84.000 | 0.881 | 8.882 | -123.342 | 115.498 |
| 85.000 | 0.478 | 8.688 | -122.995 | 123.116 |
| 86.000 | 0.050 | 8.328 | -122.580 | 130.361 |
| 87.000 | -0.396 | 7.768 | -122.130 | 137.179 |
| 88.000 | -0.860 | 6.998 | -121.649 | 143.556 |
| 89.000 | -1.372 | 6.066 | -121.139 | 149.490 |
| 90.000 | -1.917 | 4.907 | -120.607 | 154.975 |
| 91.000 | -2.553 | 6.360 | -119.561 | 149.359 |
| 92.000 | -2.723 | 3.219 | -119.100 | 154.484 |
| 93.000 | -3.628 | -1.867 | -118.570 | 159.414 |
| 94.000 | -4.930 | -2.807 | -117.988 | 163.773 |
| 95.000 | -5.342 | -2.113 | -119.006 | 14.223 |
| 96.000 | -5.136 | -7.017 | -119.575 | 11.252 |
| 97.000 | -5.721 | -14.781 | -120.251 | 7.634 |
| 98.000 | -6.789 | -17.769 | -120.990 | 3.514 |
| 99.000 | -6.998 | -17.620 | -121.738 | -1.495 |
| 100.000 | -6.706 | -23.629 | -122.498 | -7.952 |
| 101.000 | -7.180 | -33.313 | -123.265 | -15.983 |
| 102.000 | -8.206 | -38.625 | -123.962 | -25.286 |
| 103.000 | -8.414 | -40.311 | -124.421 | -35.901 |
| 104.000 | -8.031 | -47.854 | -124.568 | -47.619 |
| 105.000 | -8.375 | -59.257 | -124.432 | -59.398 |
| 106.000 | -9.279 | -65.931 | -124.062 | -69.938 |
| 107.000 | -9.243 | -68.409 | -123.514 | -78.792 |
| 108.000 | -8.531 | -77.667 | -122.914 | -86.314 |
| 109.000 | -8.764 | -91.749 | -122.402 | -92.715 |
| 110.000 | -10.037 | -99.590 | -122.080 | -97.749 |
| 111.000 | -10.157 | -97.850 | -121.902 | -101.743 |
| 112.000 | -8.716 | -105.381 | -121.907 | -105.560 |
| 113.000 | -8.371 | -124.184 | -122.205 | -109.706 |

(TABLE 2 - CONTINUED)

| | | | |
|---|---|---|---|
| 114.000 | -10.167 | -142.012 | -122.904 | -113.894 |
| 115.000 | -12.893 | -140.526 | -123.966 | -118.228 |
| 116.000 | -11.275 | -132.610 | -125.354 | -124.327 |
| 117.000 | -8.882 | -152.178 | -127.265 | -135.534 |
| 118.000 | -8.674 | 176.150 | -129.672 | -156.274 |
| 119.000 | -11.018 | 144.553 | -131.250 | 170.848 |
| 120.000 | -16.360 | 136.975 | -130.103 | 139.199 |
| 121.000 | -14.048 | 168.498 | -127.669 | 121.204 |
| 122.000 | -8.873 | 148.539 | -125.472 | 111.054 |
| 123.000 | -6.900 | 115.103 | -124.009 | 103.660 |
| 124.000 | -7.629 | 81.177 | -123.420 | 97.961 |
| 125.000 | -11.642 | 55.124 | -123.611 | 94.174 |
| 126.000 | -17.757 | 80.400 | -124.398 | 91.478 |
| 127.000 | -11.696 | 97.430 | -125.828 | 87.092 |
| 128.000 | -8.319 | 66.666 | -128.393 | 76.540 |
| 129.000 | -7.550 | 26.988 | -132.577 | 48.681 |
| 130.000 | -8.407 | -13.563 | -133.668 | -11.746 |
| 131.000 | -10.847 | -46.476 | -129.444 | -46.544 |
| 132.000 | -14.359 | -51.909 | -126.347 | -57.913 |
| 133.000 | -12.254 | -36.506 | -124.508 | -62.648 |
| 134.000 | -8.573 | -52.061 | -123.699 | -66.192 |
| 135.000 | -7.354 | -80.123 | -124.016 | -70.324 |
| 136.000 | -8.537 | -111.299 | -125.739 | -75.666 |
| 137.000 | -12.240 | -139.766 | -129.443 | -84.774 |
| 138.000 | -17.524 | -141.113 | -136.273 | -118.016 |
| 139.000 | -13.566 | -126.662 | -134.006 | 159.660 |
| 140.000 | -7.798 | -154.211 | -127.847 | 138.908 |
| 141.000 | -4.055 | 170.506 | -124.578 | 131.720 |
| 142.000 | -2.084 | 134.759 | -123.171 | 126.337 |
| 143.000 | -1.716 | 97.941 | -123.434 | 120.790 |
| 144.000 | -2.773 | 57.522 | -125.637 | 114.372 |
| 145.000 | -4.811 | 10.509 | -130.873 | 102.363 |
| 146.000 | -7.185 | -42.313 | -139.789 | 25.751 |
| 147.000 | -10.738 | -97.049 | -130.656 | -36.306 |
| 148.000 | -18.547 | 166.132 | -125.977 | -41.811 |
| 149.000 | -9.374 | 52.100 | -123.711 | -42.512 |
| 150.000 | -3.399 | 9.179 | -123.007 | -43.618 |
| 151.000 | -0.622 | -27.103 | -123.981 | -47.454 |
| 152.000 | 0.511 | -64.930 | -127.491 | -58.417 |
| 153.000 | 1.028 | -106.879 | -135.410 | -112.772 |
| 154.000 | 1.773 | -150.367 | -128.474 | 164.551 |
| 155.000 | 2.655 | 169.735 | -123.284 | 148.255 |
| 156.000 | 2.982 | 133.899 | -121.297 | 140.515 |
| 157.000 | 2.346 | 98.230 | -121.502 | 134.297 |
| 158.000 | 0.955 | 56.745 | -124.004 | 126.351 |
| 159.000 | 0.169 | 6.055 | -130.912 | 105.643 |
| 160.000 | 1.100 | -43.307 | -132.447 | -2.658 |
| 161.000 | 2.352 | -83.833 | -124.636 | -28.435 |
| 162.000 | 3.025 | -121.222 | -121.514 | -35.673 |
| 163.000 | 3.603 | -160.794 | -120.732 | -37.259 |
| 164.000 | 3.121 | 159.640 | -122.514 | -40.537 |
| 165.000 | 3.186 | 121.981 | -127.833 | -41.935 |
| 166.000 | 3.582 | 91.819 | -149.018 | 159.574 |
| 167.000 | 3.284 | 67.246 | -126.600 | 145.894 |
| 168.000 | 1.840 | 42.724 | -121.729 | 145.149 |
| 169.000 | -0.566 | 9.675 | -120.158 | 146.036 |
| 170.000 | -1.587 | -38.024 | -120.284 | 148.279 |
| 171.000 | 0.356 | -77.615 | -123.674 | 150.846 |
| 172.000 | 2.090 | -100.324 | -135.072 | 156.949 |
| 173.000 | 2.296 | -115.463 | -126.944 | -33.554 |
| 174.000 | 0.664 | -132.219 | -128.067 | 13.192 |
| 175.000 | -2.889 | -161.415 | -121.323 | 65.996 |
| 176.000 | -3.472 | 137.785 | -101.095 | -75.063 |
| 177.000 | 1.231 | 102.703 | -111.180 | -79.016 |
| 178.000 | 4.529 | 91.185 | -100.571 | -103.429 |
| 179.000 | 6.297 | 86.110 | -91.446 | 58.021 |

(TABLE 2 - CONTINUED)

| | | | | |
|---|---|---|---|---|
| 180.000 | 6.886 | 84.371 | -113.952 | 37.563 |
| 0.000 | 0.967 | 8.395 | -123.048 | -63.194 |
| 1.000 | 1.143 | 7.866 | -38.904 | 157.565 |
| 2.000 | 1.641 | 6.510 | -27.562 | 159.050 |
| 3.000 | 2.331 | 4.825 | -21.518 | 159.313 |
| 4.000 | 3.054 | 3.313 | -17.959 | 159.588 |
| 5.000 | 3.679 | 2.256 | -16.041 | 159.893 |
| 6.000 | 4.121 | 1.739 | -15.457 | 160.273 |
| 7.000 | 4.453 | 0.130 | -17.175 | 158.870 |
| 8.000 | 4.440 | 0.398 | -17.402 | 159.474 |
| 9.000 | 4.220 | 1.176 | -20.014 | 160.257 |
| 10.000 | 3.857 | 2.275 | -26.958 | 162.007 |
| 11.000 | 3.436 | 3.437 | -36.132 | -25.388 |
| 12.000 | 3.052 | 4.356 | -23.486 | -20.141 |
| 13.000 | 2.798 | 4.759 | -20.143 | -18.862 |
| 14.000 | 2.732 | 4.534 | -19.639 | -17.785 |
| 15.000 | 2.858 | 3.800 | -21.386 | -16.544 |
| 16.000 | 3.127 | 2.843 | -26.446 | -14.461 |
| 17.000 | 3.465 | 1.977 | -52.327 | 44.052 |
| 18.000 | 3.786 | 1.408 | -27.372 | 162.220 |
| 19.000 | 4.021 | 1.226 | -22.550 | 164.595 |
| 20.000 | 4.127 | 1.409 | -21.123 | 166.195 |
| 21.000 | 4.091 | 1.875 | -21.842 | 167.810 |
| 22.000 | 3.925 | 2.497 | -24.973 | 169.936 |
| 23.000 | 3.656 | 3.123 | -33.629 | 175.639 |
| 24.000 | 3.349 | 3.597 | -34.860 | -17.874 |
| 25.000 | 3.066 | 3.783 | -25.827 | -11.242 |
| 26.000 | 2.869 | 3.619 | -22.823 | -8.811 |
| 27.000 | 2.802 | 3.152 | -22.190 | -6.789 |
| 28.000 | 2.876 | 2.534 | -23.449 | -4.626 |
| 29.000 | 3.065 | 1.956 | -27.165 | -1.577 |
| 30.000 | 3.312 | 1.574 | -37.803 | 9.430 |
| 31.000 | 3.568 | 1.469 | -33.920 | 169.660 |
| 32.000 | 3.782 | 1.640 | -26.432 | 176.327 |
| 33.000 | 3.917 | 2.025 | -23.665 | 179.261 |
| 34.000 | 3.953 | 2.533 | -22.956 | -178.282 |
| 35.000 | 3.886 | 3.053 | -23.878 | -175.732 |
| 36.000 | 3.725 | 3.478 | -26.730 | -172.471 |
| 37.000 | 3.491 | 3.708 | -33.525 | -165.437 |
| 38.000 | 3.228 | 3.670 | -42.176 | -18.272 |
| 39.000 | 2.977 | 3.340 | -29.614 | 3.649 |
| 40.000 | 2.780 | 2.760 | -25.507 | 8.068 |
| 41.000 | 2.670 | 2.038 | -23.814 | 11.125 |
| 42.000 | 2.660 | 1.332 | -23.586 | 13.933 |
| 43.000 | 2.734 | 0.796 | -24.668 | 16.926 |
| 44.000 | 2.878 | 0.539 | -27.353 | 20.705 |
| 45.000 | 3.063 | 0.605 | -33.075 | 27.920 |
| 46.000 | 3.257 | 0.977 | -47.750 | 128.511 |
| 47.000 | 3.427 | 1.581 | -32.425 | -166.778 |
| 48.000 | 3.553 | 2.339 | -27.395 | -160.205 |
| 49.000 | 3.614 | 3.149 | -25.019 | -156.375 |
| 50.000 | 3.601 | 3.921 | -24.018 | -153.154 |
| 51.000 | 3.522 | 4.561 | -24.019 | -150.050 |
| 52.000 | 3.213 | 5.002 | -29.394 | -121.519 |
| 53.000 | 3.033 | 5.483 | -29.406 | -101.627 |
| 54.000 | 2.835 | 5.620 | -29.416 | -82.155 |
| 55.000 | 2.629 | 5.394 | -29.822 | -63.056 |
| 56.000 | 2.432 | 4.817 | -29.425 | -44.351 |
| 57.000 | 2.271 | 3.936 | -29.428 | -26.068 |
| 58.000 | 2.160 | 2.848 | -29.430 | -8.191 |
| 59.000 | 2.106 | 1.676 | -29.424 | 9.275 |
| 60.000 | 2.110 | 0.551 | -29.421 | 26.318 |
| 61.000 | 2.149 | -0.401 | -29.416 | 42.945 |
| 62.000 | 2.222 | -1.106 | -29.407 | 59.134 |
| 63.000 | 2.319 | -1.516 | -29.395 | 74.879 |
| 64.000 | 2.425 | -1.617 | -29.382 | 90.192 |

(TABLE 2 - CONTINUED)

| | | | | |
|---|---|---|---|---|
| 65.000 | 2.530 | -1.423 | -29.366 | 105.068 |
| 66.000 | 2.613 | -0.955 | -29.348 | 119.498 |
| 67.000 | 2.662 | -0.254 | -29.329 | 133.451 |
| 68.000 | 2.686 | 0.627 | -29.308 | 146.951 |
| 69.000 | 2.680 | 1.643 | -29.285 | 159.988 |
| 70.000 | 2.643 | 2.750 | -29.258 | 172.558 |
| 71.000 | 2.568 | 3.920 | -29.230 | -175.348 |
| 72.000 | 2.445 | 5.136 | -29.200 | -163.733 |
| 73.000 | 2.292 | 6.342 | -29.169 | -152.602 |
| 74.000 | 2.110 | 7.511 | -29.136 | -141.960 |
| 75.000 | 1.899 | 8.618 | -29.101 | -131.810 |
| 76.000 | 1.651 | 9.682 | -29.065 | -122.154 |
| 77.000 | 1.364 | 10.691 | -29.027 | -112.998 |
| 78.000 | 1.054 | 11.595 | -28.988 | -104.345 |
| 79.000 | 0.723 | 12.381 | -28.948 | -96.200 |
| 80.000 | 0.372 | 13.038 | -28.905 | -88.569 |
| 81.000 | -0.028 | 13.687 | -28.862 | -81.457 |
| 82.000 | -0.448 | 14.202 | -28.816 | -74.868 |
| 83.000 | -0.886 | 14.576 | -28.768 | -68.809 |
| 84.000 | -1.341 | 14.801 | -28.719 | -63.283 |
| 85.000 | -1.844 | 15.028 | -28.667 | -58.296 |
| 86.000 | -2.380 | 15.164 | -28.613 | -53.847 |
| 87.000 | -2.938 | 15.144 | -28.555 | -49.950 |
| 88.000 | -12.344 | 164.489 | -28.494 | -46.611 |
| 89.000 | -11.897 | 169.730 | -28.428 | -43.828 |
| 90.000 | -11.445 | 174.602 | -28.356 | -41.609 |
| 91.000 | -11.016 | -175.870 | -19.497 | 121.243 |
| 92.000 | -10.541 | -171.972 | -16.802 | 38.902 |
| 93.000 | -10.073 | -168.410 | -17.259 | -36.145 |
| 94.000 | -9.611 | -165.177 | -19.599 | -129.047 |
| 95.000 | -9.277 | 7.385 | -18.958 | 119.218 |
| 96.000 | -9.632 | 4.007 | -16.070 | 38.164 |
| 97.000 | -9.971 | 0.427 | -16.373 | -36.418 |
| 98.000 | -10.297 | -3.324 | -18.501 | -125.550 |
| 99.000 | -10.616 | -7.217 | -18.497 | 130.934 |
| 100.000 | -10.935 | -11.231 | -16.773 | 39.452 |
| 101.000 | -11.264 | -15.358 | -15.781 | -42.951 |
| 102.000 | -11.614 | -19.608 | -16.029 | -122.128 |
| 103.000 | -11.999 | -24.011 | -18.143 | 152.054 |
| 104.000 | -12.435 | -28.629 | -18.733 | 41.265 |
| 105.000 | -12.934 | -33.559 | -14.999 | -48.698 |
| 106.000 | -13.511 | -38.947 | -13.771 | -114.560 |
| 107.000 | -14.174 | -44.999 | -17.045 | -179.689 |
| 108.000 | -14.926 | -51.997 | -23.175 | 47.780 |
| 109.000 | -15.747 | -60.301 | -14.840 | -50.512 |
| 110.000 | -16.589 | -70.308 | -12.586 | -104.933 |
| 111.000 | -17.350 | -82.296 | -15.572 | -155.503 |
| 112.000 | -17.886 | -96.114 | -31.347 | 100.294 |
| 113.000 | -18.070 | -110.881 | -17.048 | -46.951 |
| 114.000 | -17.887 | -125.209 | -13.958 | -95.652 |
| 115.000 | -17.464 | -137.980 | -16.304 | -143.488 |
| 116.000 | -16.981 | -148.825 | -26.158 | 153.879 |
| 117.000 | -16.594 | -157.980 | -26.516 | -29.720 |
| 118.000 | -16.409 | -165.950 | -22.530 | -102.627 |
| 119.000 | -16.499 | -173.324 | -22.208 | 176.662 |
| 120.000 | -16.923 | 179.232 | -21.879 | 109.879 |
| 121.000 | -17.727 | 170.867 | -27.907 | 89.702 |
| 122.000 | -18.943 | 160.299 | -20.673 | 153.452 |
| 123.000 | -20.494 | 145.445 | -13.977 | 123.793 |
| 124.000 | -21.953 | 123.765 | -12.873 | 84.922 |
| 125.000 | -22.359 | 96.297 | -17.393 | 37.869 |
| 126.000 | -21.360 | 71.195 | -23.943 | -125.896 |
| 127.000 | -19.886 | 53.391 | -13.125 | 161.027 |
| 128.000 | -18.710 | 41.218 | -10.367 | 107.371 |
| 129.000 | -18.097 | 32.077 | -10.012 | 45.000 |
| 130.000 | -18.138 | 24.074 | -9.434 | -23.032 |

(TABLE 2 - CONTINUED)

| | | | |
|---|---|---|---|
| 131.000 | -18.908 | 15.572 | -9.109 | -83.997 |
| 132.000 | -20.516 | 4.273 | -11.301 | -144.182 |
| 133.000 | -22.958 | -14.596 | -15.040 | 119.867 |
| 134.000 | -25.067 | -49.050 | -10.014 | 26.587 |
| 135.000 | -24.005 | -89.187 | -6.675 | -26.338 |
| 136.000 | -21.406 | -113.440 | -6.420 | -73.276 |
| 137.000 | -19.516 | -126.700 | -8.311 | -125.962 |
| 138.000 | -18.664 | -135.565 | -10.587 | 170.781 |
| 139.000 | -18.865 | -143.285 | -12.463 | 111.840 |
| 140.000 | -20.237 | -152.437 | -18.950 | 69.393 |
| 141.000 | -23.058 | -167.949 | -19.962 | -162.909 |
| 142.000 | -26.790 | 155.230 | -10.157 | 166.680 |
| 143.000 | -25.593 | 99.698 | -7.625 | 130.548 |
| 144.000 | -21.774 | 72.879 | -8.386 | 86.059 |
| 145.000 | -19.455 | 60.856 | -10.291 | 21.468 |
| 146.000 | -18.648 | 53.193 | -9.322 | -50.572 |
| 147.000 | -19.273 | 46.076 | -8.298 | -107.620 |
| 148.000 | -21.603 | 35.912 | -9.148 | -171.024 |
| 149.000 | -26.327 | 10.905 | -7.247 | 109.496 |
| 150.000 | -28.144 | -60.304 | -2.835 | 53.305 |
| 151.000 | -22.661 | -98.779 | -0.060 | 12.541 |
| 152.000 | -19.454 | -111.230 | 1.071 | -26.026 |
| 153.000 | -18.290 | -118.061 | 1.348 | -68.206 |
| 154.000 | -18.873 | -124.054 | 1.832 | -113.652 |
| 155.000 | -21.564 | -132.927 | 2.787 | -155.895 |
| 156.000 | -27.820 | -160.568 | 3.396 | 167.843 |
| 157.000 | -27.697 | 107.218 | 2.998 | 134.388 |
| 158.000 | -21.131 | 79.684 | 1.428 | 98.215 |
| 159.000 | -18.196 | 71.397 | -0.686 | 52.274 |
| 160.000 | -17.502 | 66.474 | -1.413 | -2.440 |
| 161.000 | -18.863 | 61.436 | -0.771 | -51.378 |
| 162.000 | -23.280 | 51.187 | -0.501 | -96.343 |
| 163.000 | -32.740 | -22.193 | 0.036 | -147.008 |
| 164.000 | -22.751 | -93.849 | -0.261 | 168.168 |
| 165.000 | -17.927 | -103.490 | 0.340 | 128.896 |
| 166.000 | -16.203 | -107.588 | 1.176 | 100.487 |
| 167.000 | -16.688 | -110.831 | 0.989 | 78.534 |
| 168.000 | -19.926 | -115.819 | -0.656 | 56.283 |
| 169.000 | -30.526 | -143.949 | -3.636 | 22.930 |
| 170.000 | -23.418 | 87.685 | -4.605 | -31.198 |
| 171.000 | -16.655 | 78.274 | -1.747 | -70.944 |
| 172.000 | -14.057 | 75.350 | 0.335 | -90.787 |
| 173.000 | -13.824 | 73.536 | 0.602 | -103.521 |
| 174.000 | -12.167 | 75.921 | -2.290 | -114.182 |
| 175.000 | -12.749 | 75.249 | -9.559 | -148.402 |
| 176.000 | -14.669 | 74.782 | -6.047 | 109.295 |
| 177.000 | -18.225 | 74.437 | 1.039 | 91.209 |
| 178.000 | -24.278 | 74.069 | 4.524 | 87.007 |
| 179.000 | -35.629 | 72.724 | 6.287 | 85.284 |
| 180.000 | -126.056 | -140.389 | 6.886 | 84.371 |
| 0.000 | 0.967 | 8.395 | -123.560 | -71.142 |
| 1.000 | 1.210 | 7.958 | -108.371 | 44.911 |
| 2.000 | 1.861 | 6.773 | -108.842 | -97.689 |
| 3.000 | 2.718 | 5.288 | -119.681 | -2.060 |
| 4.000 | 3.577 | 3.892 | -126.287 | -23.311 |
| 5.000 | 4.281 | 2.773 | -133.494 | -69.934 |
| 6.000 | 4.739 | 1.983 | -126.804 | -44.951 |
| 7.000 | 4.653 | 3.521 | -134.996 | 11.195 |
| 8.000 | 4.618 | 3.677 | -134.950 | 2.367 |
| 9.000 | 4.333 | 3.636 | -135.487 | -6.151 |
| 10.000 | 3.876 | 3.377 | -136.588 | -14.116 |
| 11.000 | 3.349 | 2.914 | -138.240 | -20.795 |
| 12.000 | 2.878 | 2.319 | -140.439 | -24.253 |
| 13.000 | 2.581 | 1.743 | -143.030 | -20.444 |
| 14.000 | 2.530 | 1.376 | -144.761 | -4.316 |
| 15.000 | 2.714 | 1.343 | -143.809 | 15.470 |

(TABLE 2 - CONTINUED)

| | | | | |
|---|---|---|---|---|
| 16.000 | 3.056 | 1.651 | -141.367 | 24.313 |
| 17.000 | 3.459 | 2.198 | -139.124 | 24.548 |
| 18.000 | 3.821 | 2.839 | -137.477 | 20.746 |
| 19.000 | 4.067 | 3.444 | -136.424 | 15.226 |
| 20.000 | 4.157 | 3.898 | -135.912 | 8.941 |
| 21.000 | 4.082 | 4.118 | -135.898 | 2.312 |
| 22.000 | 3.862 | 4.040 | -136.341 | -4.418 |
| 23.000 | 3.535 | 3.648 | -137.212 | -10.839 |
| 24.000 | 3.179 | 2.974 | -138.470 | -16.312 |
| 25.000 | 2.870 | 2.143 | -140.064 | -19.604 |
| 26.000 | 2.676 | 1.356 | -141.846 | -18.521 |
| 27.000 | 2.633 | 0.837 | -143.340 | -10.494 |
| 28.000 | 2.740 | 0.742 | -143.623 | 3.359 |
| 29.000 | 2.955 | 1.098 | -142.431 | 15.375 |
| 30.000 | 3.209 | 1.814 | -140.674 | 21.092 |
| 31.000 | 3.457 | 2.727 | -139.058 | 21.847 |
| 32.000 | 3.648 | 3.663 | -137.819 | 19.738 |
| 33.000 | 3.748 | 4.465 | -136.994 | 16.040 |
| 34.000 | 3.744 | 5.006 | -136.564 | 11.388 |
| 35.000 | 3.636 | 5.195 | -136.481 | 6.137 |
| 36.000 | 3.439 | 4.984 | -136.715 | 0.529 |
| 37.000 | 3.180 | 4.376 | -137.230 | -5.210 |
| 38.000 | 2.910 | 3.430 | -137.980 | -10.746 |
| 39.000 | 2.673 | 2.289 | -138.928 | -15.558 |
| 40.000 | 2.505 | 1.162 | -140.035 | -18.869 |
| 41.000 | 2.427 | 0.275 | -141.242 | -19.607 |
| 42.000 | 2.438 | -0.191 | -142.407 | -16.565 |
| 43.000 | 2.509 | -0.141 | -143.254 | -9.104 |
| 44.000 | 2.624 | 0.409 | -143.394 | 1.416 |
| 45.000 | 2.760 | 1.356 | -142.737 | 11.467 |
| 46.000 | 2.889 | 2.560 | -141.628 | 18.422 |
| 47.000 | 2.989 | 3.855 | -140.459 | 22.011 |
| 48.000 | 3.045 | 5.099 | -139.441 | 23.043 |
| 49.000 | 3.036 | 6.173 | -138.658 | 22.328 |
| 50.000 | 2.960 | 6.978 | -138.115 | 20.402 |
| 51.000 | 2.832 | 7.422 | -137.787 | 17.568 |
| 52.000 | 2.441 | 6.683 | -140.084 | 6.671 |
| 53.000 | 2.213 | 6.715 | -140.315 | 6.689 |
| 54.000 | 1.994 | 6.369 | -140.536 | 6.330 |
| 55.000 | 1.793 | 5.705 | -140.739 | 5.655 |
| 56.000 | 1.624 | 4.806 | -140.909 | 4.745 |
| 57.000 | 1.511 | 3.768 | -141.024 | 3.702 |
| 58.000 | 1.456 | 2.726 | -141.079 | 2.661 |
| 59.000 | 1.455 | 1.808 | -141.082 | 1.751 |
| 60.000 | 1.496 | 1.116 | -141.042 | 1.073 |
| 61.000 | 1.545 | 0.745 | -140.995 | 0.721 |
| 62.000 | 1.601 | 0.693 | -140.943 | 0.693 |
| 63.000 | 1.657 | 0.946 | -140.891 | 0.971 |
| 64.000 | 1.701 | 1.477 | -140.853 | 1.527 |
| 65.000 | 1.725 | 2.241 | -140.835 | 2.316 |
| 66.000 | 1.711 | 3.219 | -140.857 | 3.315 |
| 67.000 | 1.649 | 4.379 | -140.928 | 4.494 |
| 68.000 | 1.554 | 5.634 | -141.032 | 5.763 |
| 69.000 | 1.427 | 6.941 | -141.170 | 7.079 |
| 70.000 | 1.267 | 8.261 | -141.341 | 8.401 |
| 71.000 | 1.066 | 9.595 | -141.554 | 9.732 |
| 72.000 | 0.811 | 10.973 | -141.820 | 11.099 |
| 73.000 | 0.529 | 12.283 | -142.115 | 12.389 |
| 74.000 | 0.220 | 13.503 | -142.437 | 13.580 |
| 75.000 | -0.113 | 14.612 | -142.782 | 14.652 |
| 76.000 | -0.486 | 15.711 | -143.168 | 15.701 |
| 77.000 | -0.899 | 16.802 | -143.593 | 16.732 |
| 78.000 | -1.330 | 17.761 | -144.037 | 17.618 |
| 79.000 | -1.778 | 18.581 | -144.495 | 18.352 |
| 80.000 | -2.240 | 19.253 | -144.967 | 18.925 |
| 81.000 | -2.763 | 20.180 | -145.498 | 19.738 |

(TABLE 2 - CONTINUED)

| | | | | |
|---|---|---|---|---|
| 82.000 | -3.302 | 20.986 | -146.044 | 20.413 |
| 83.000 | -3.856 | 21.659 | -146.603 | 20.938 |
| 84.000 | -4.426 | 22.192 | -147.174 | 21.306 |
| 85.000 | -5.052 | 23.127 | -147.800 | 22.052 |
| 86.000 | -5.717 | 24.163 | -148.461 | 22.874 |
| 87.000 | -6.410 | 25.098 | -149.146 | 23.563 |
| 88.000 | -7.137 | 25.929 | -149.859 | 24.113 |
| 89.000 | -7.934 | 27.678 | -150.640 | 25.524 |
| 90.000 | -8.797 | 29.780 | -151.482 | 27.216 |
| 91.000 | -6.894 | 0.484 | -148.344 | -16.725 |
| 92.000 | -18.200 | -11.106 | -148.576 | -16.249 |
| 93.000 | -10.874 | 77.345 | -148.892 | -16.462 |
| 94.000 | -7.008 | 41.726 | -149.657 | -16.291 |
| 95.000 | -9.582 | -4.731 | -149.645 | -8.230 |
| 96.000 | -23.772 | -81.953 | -148.810 | -5.792 |
| 97.000 | -11.906 | 93.435 | -148.414 | -8.293 |
| 98.000 | -7.499 | 39.802 | -149.023 | -7.558 |
| 99.000 | -8.685 | -13.869 | -149.105 | 1.380 |
| 100.000 | -17.197 | -89.486 | -148.004 | 5.063 |
| 101.000 | -13.145 | 102.643 | -147.904 | 1.059 |
| 102.000 | -7.699 | 39.587 | -149.391 | 3.439 |
| 103.000 | -7.980 | -17.712 | -149.688 | 18.939 |
| 104.000 | -13.526 | -92.381 | -148.304 | 22.707 |
| 105.000 | -13.795 | 124.896 | -148.867 | 16.127 |
| 106.000 | -8.428 | 47.382 | -152.028 | 22.285 |
| 107.000 | -8.046 | -16.087 | -152.133 | 51.304 |
| 108.000 | -11.181 | -92.745 | -150.251 | 52.323 |
| 109.000 | -12.637 | 158.138 | -152.425 | 42.005 |
| 110.000 | -10.031 | 68.311 | -159.654 | 73.363 |
| 111.000 | -9.498 | -7.565 | -154.119 | 119.383 |
| 112.000 | -10.202 | -88.751 | -152.643 | 108.530 |
| 113.000 | -10.592 | -171.804 | -158.388 | 113.013 |
| 114.000 | -11.775 | 106.971 | -155.670 | -172.593 |
| 115.000 | -13.215 | 12.420 | -150.573 | -177.906 |
| 116.000 | -11.403 | -78.644 | -151.225 | 172.105 |
| 117.000 | -10.477 | -146.432 | -154.686 | -165.454 |
| 118.000 | -13.337 | 150.701 | -151.474 | -134.857 |
| 119.000 | -20.425 | 48.160 | -149.593 | -139.022 |
| 120.000 | -16.017 | -66.398 | -151.620 | -141.711 |
| 121.000 | -15.061 | -125.226 | -154.447 | -116.226 |
| 122.000 | -22.027 | 176.509 | -153.209 | -96.305 |
| 123.000 | -23.835 | -14.532 | -154.992 | -93.396 |
| 124.000 | -17.874 | -80.593 | -159.169 | -48.148 |
| 125.000 | -19.409 | -168.018 | -154.261 | -18.339 |
| 126.000 | -14.850 | 84.520 | -154.579 | -24.324 |
| 127.000 | -10.714 | 17.928 | -162.049 | 28.088 |
| 128.000 | -10.145 | -47.197 | -151.447 | 69.991 |
| 129.000 | -10.061 | -129.731 | -147.878 | 50.578 |
| 130.000 | -7.329 | 151.685 | -149.828 | 23.129 |
| 131.000 | -5.465 | 88.954 | -164.406 | -38.737 |
| 132.000 | -5.620 | 25.468 | -151.496 | 161.625 |
| 133.000 | -5.886 | -50.035 | -147.536 | 125.276 |
| 134.000 | -4.199 | -123.032 | -149.294 | 80.832 |
| 135.000 | -2.897 | 176.848 | -152.505 | -9.266 |
| 136.000 | -3.216 | 118.732 | -147.352 | -85.919 |
| 137.000 | -4.208 | 52.928 | -145.211 | -127.239 |
| 138.000 | -4.089 | -15.320 | -147.188 | -164.968 |
| 139.000 | -3.911 | -74.246 | -154.608 | 133.344 |
| 140.000 | -5.456 | -130.236 | -152.638 | 12.120 |
| 141.000 | -8.351 | 159.964 | -149.009 | -35.315 |
| 142.000 | -8.361 | 78.855 | -151.072 | -76.168 |
| 143.000 | -7.437 | 18.387 | -157.705 | -160.880 |
| 144.000 | -9.546 | -36.077 | -152.278 | 104.122 |
| 145.000 | -14.118 | -123.312 | -150.306 | 63.331 |
| 146.000 | -10.246 | 139.402 | -154.792 | 21.759 |
| 147.000 | -7.191 | 79.624 | -159.212 | -130.381 |

(TABLE 2 - CONTINUED)

| | | | | |
|---|---|---|---|---|
| 148.000 | -5.950 | 16.532 | -148.652 | 173.045 |
| 149.000 | -3.189 | -49.164 | -145.300 | 132.897 |
| 150.000 | 0.210 | -100.484 | -143.992 | 85.835 |
| 151.000 | 2.431 | -142.558 | -142.335 | 33.572 |
| 152.000 | 3.450 | 176.693 | -140.091 | -12.930 |
| 153.000 | 3.908 | 133.344 | -138.407 | -52.429 |
| 154.000 | 4.568 | 88.690 | -137.622 | -89.613 |
| 155.000 | 5.464 | 47.481 | -137.447 | -127.713 |
| 156.000 | 5.985 | 11.041 | -137.452 | -166.868 |
| 157.000 | 5.714 | -23.198 | -137.460 | 154.887 |
| 158.000 | 4.637 | -58.896 | -137.749 | 118.969 |
| 159.000 | 3.189 | -98.981 | -138.500 | 85.279 |
| 160.000 | 2.029 | -142.197 | -140.106 | 49.791 |
| 161.000 | 0.969 | 176.041 | -142.455 | 10.842 |
| 162.000 | -1.089 | 132.861 | -145.656 | -36.659 |
| 163.000 | -3.617 | 68.768 | -148.418 | -108.961 |
| 164.000 | -3.954 | 14.578 | -143.697 | -157.234 |
| 165.000 | -2.785 | -25.604 | -144.035 | 166.474 |
| 166.000 | -1.611 | -52.040 | -143.844 | 134.127 |
| 167.000 | -1.739 | -71.742 | -143.912 | 106.757 |
| 168.000 | -3.557 | -92.097 | -146.741 | 81.996 |
| 169.000 | -7.069 | -125.711 | -148.707 | 51.470 |
| 170.000 | -7.892 | 173.963 | -149.858 | 7.639 |
| 171.000 | -4.239 | 135.206 | -148.471 | -33.268 |
| 172.000 | -1.952 | 117.796 | -146.925 | -59.668 |
| 173.000 | -1.660 | 106.856 | -146.352 | -80.852 |
| 174.000 | -3.881 | 93.067 | -125.159 | -119.108 |
| 175.000 | -15.556 | 119.334 | -126.578 | -89.548 |
| 176.000 | -5.983 | -107.781 | -123.165 | -74.731 |
| 177.000 | 1.144 | -100.359 | -124.399 | -129.605 |
| 178.000 | 4.542 | -97.221 | -102.520 | 111.021 |
| 179.000 | 6.281 | -95.619 | -94.737 | -10.519 |
| 180.000 | 6.886 | -95.629 | -101.403 | 27.062 |

```
TO: LAIC=T, LGTD=T, LFEED=T
F    90.   1.
F    32   32   1   48
F    F    F    F    0
T    T    F    0.8
T    F    F    T    T    F    0.   0.
T    T    0.   0.
FB:
1 0. 0. 8.
2.92
PZ:
2
0. 90.
0. 180. 0.5
F
PP:
1
1   1
1   2
XQ:
```

Example 2:

This example simply illustrates the radiation patterns of an offset reflector with an elliptic rim. The length of the major and minor axis of the ellipse are DAA=12.0" and DBB=10.0". The center of the ellipse is offset along the y-axis by 5.0" and the feed is tilted by 32.0 degrees, as specified by the TL: command. The calculated patterns are shown in Figure 8. The input data are given in Table 3.

## TABLE 3

### INPUT DATA FOR THE EXAMPLE OF ELLIPTIC RIM REFLECTOR

```
CM:          *** ELRIM.DAT ***
CM: EXAMPLE OF ELLIPTIC RIM REFLECTOR
CM:       D = -1 IN DG: COMMAND
CE: YC = 5.0" , PSIT = 32.0 DEG. IN TL:
DG:   DAA = 12.0"   DBB = 10.0"
1
3   8.0   0.6   0.6    -1    0
12.  10.
TL:
32.  5.
PZ:
2
0.  90.
-180.  180.  1.
F
PP:
1
1  1
1  2
XQ:
```

(a) H-plane



(b)  E-plane
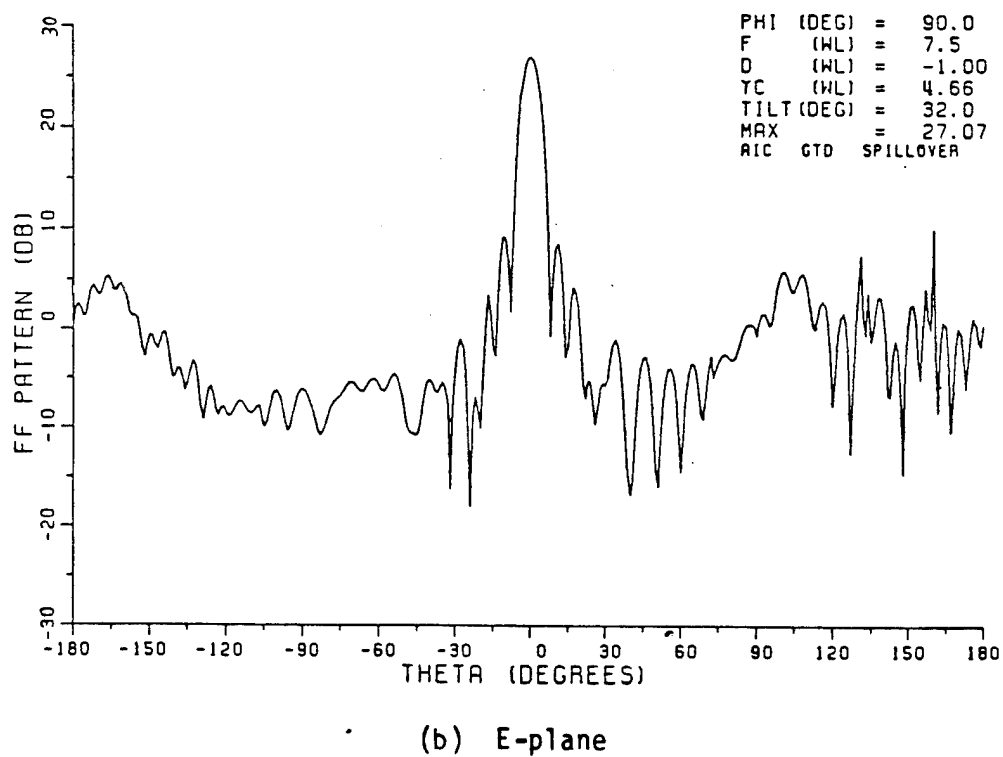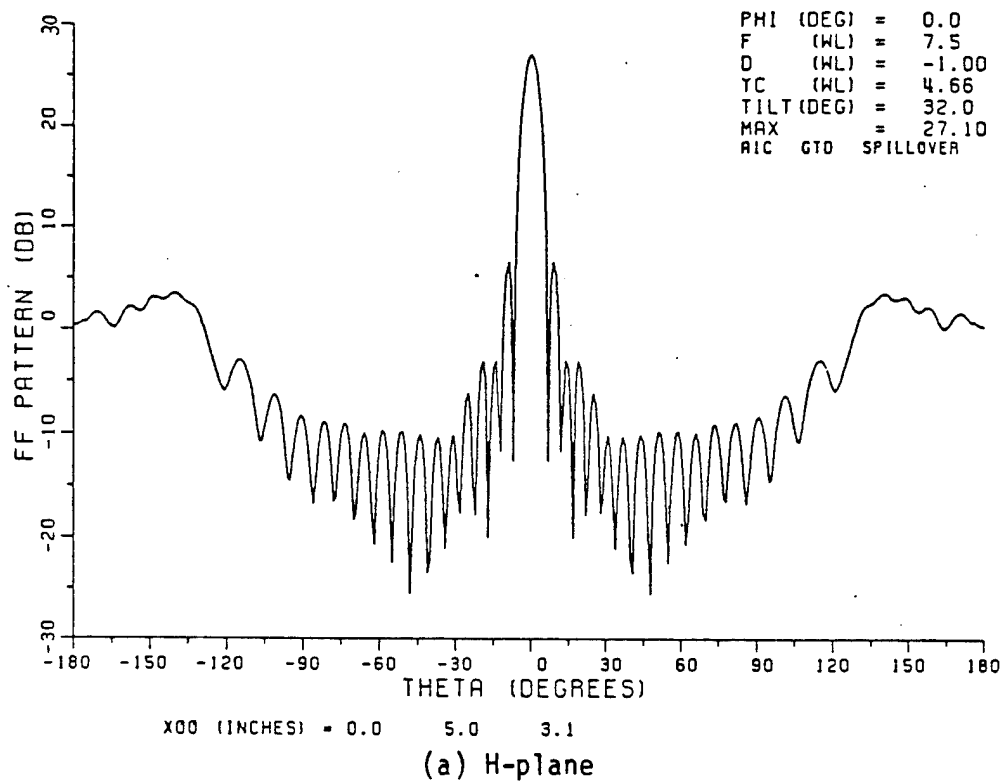
Figure 8.   Calculated patterns of an offset reflector anetnna with
elliptic rim (D=-1).

## Command AD: Extended Aperture Integration (AIE)

This command enables the user to specify the extended aperture and provides the capability for the diffracted fields to be included in the aperture fields for the Extended Aperture Integration (AIE) method [7]. This command can also be used to calculate the blockage effects by the feed structure besides using the FB: Command.

BLOCK DIAGRAM FOR AD: COMMAND

```
            | Command AD: |
                   |
                   |
            | Set LEXPR=T |
                   |
                   |
          | Read:  LAD, LRSRIM |
                   |
                   |
            | Read:  ZEX |
                   |
                   |
            | Read:  NRIMX |
                   |
    --------->| DO NE=1,NRIMX |
   |               |
   |               |
   |       | Read:  (RIMX(NE,N),N=1,2) |
   |               |
   |               |
   |               |
    --------------| Continue |
```

10-1
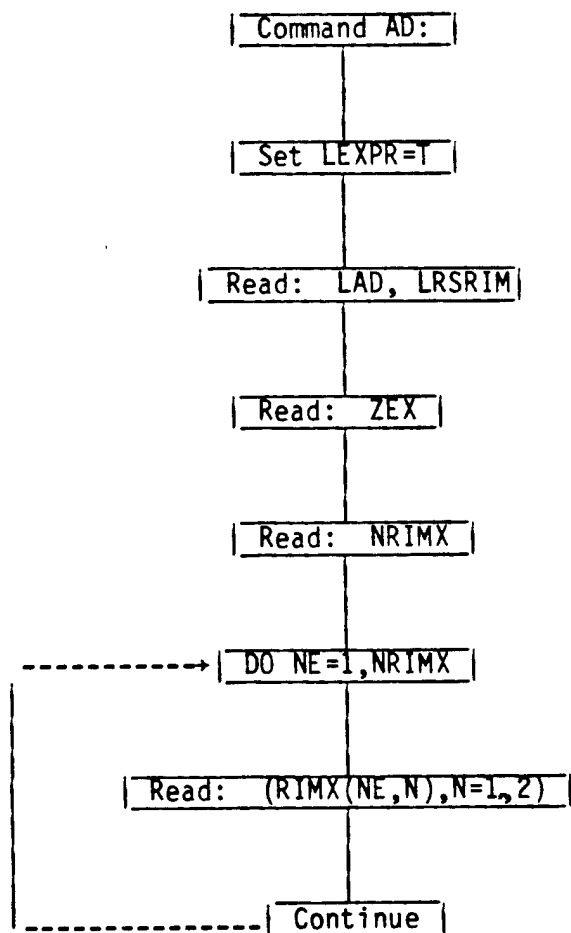
1. Read: LAD, LRSRIM

   a) LAD: This logical variable specifies whether or not the diffracted fields from the reflector rim will be added to the original G.O. aperture fields. If LAD = true, the diffracted fields will be added.

   b) LRSRIM: This logical variable specifies whether the rim points (x,y components) of the extended aperture will be set identical to the rim points of the reflector when LAD=false. If LRSRIM=true, the extended aperture rim is set identical to the reflector rim. If LRSRIM =false, the extended aperture rim is specified by NRIMX and RIMX. If LAD=true, this variable is ignored by the code.

2. Read: ZEX

   a) This real variable specifies the Z coordinate of the extended aperture as shown in Figure 1a.

      Note: The phase center for extended aperture integration is at the point (0., 0., ZEX).

3. Read: NRIMX

   a) NRIMX: This integer specifies the number of input rim points for the extended aperture.

4. Read: (RIMX(NE,N),N=1,2)

   This read statement is executed NRIMX times for each rim point.

   a) RIMX(NE,N): This doubly dimensioned real variable is used to specify the location of the NE[th] corner of the extended aperture as shown in Figure 1b. Presently, 112 rim points may be used.

      Note: The rim points RIMX(NE,N) must be input in the counterclockwise order.
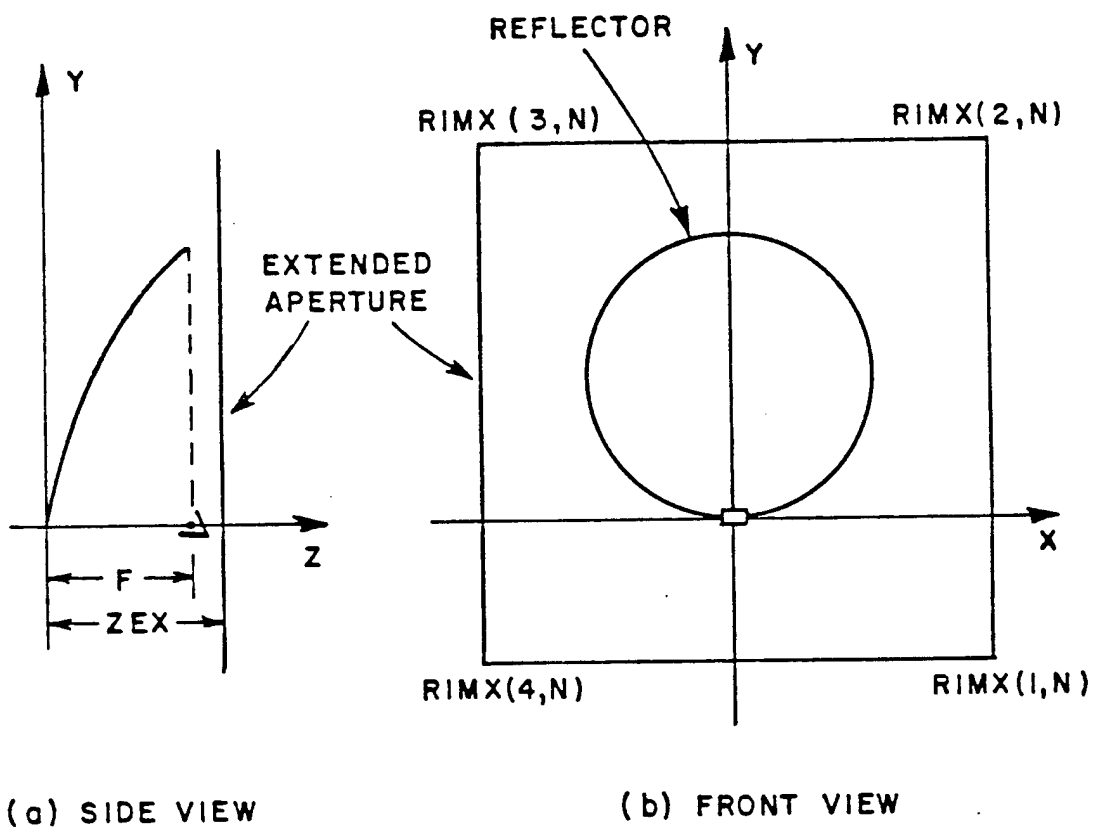
Figure 1. An example to show the position and size of the extended aperture.

Three examples are given in this section to illustrate how to use the AD: Command. The first example shows the calculation of an offset reflector pattern by the AIE method. The second and third examples show the calculation of feed blockage effects by the AD: Command.

Example 1:

This example illustrates how the extended aperture integration (AIE) is used to calculate the far-field pattern of an offset reflector. The results from multi-point GTD and the conventional aperture integration (AIC) are also given for comparison. The geometries of the reflector and the extended aperture are given in Figure 1 with ZEX=7.0". The far-field patterns for AIE, AIC and multi-point GTD are shown in Figure 2. It is found that AIE and Multi-point GTD are in good agreement at wide angles. The input data for AIE, AIC, and GTD are given in Tables 1, 2 and 3, respectively.
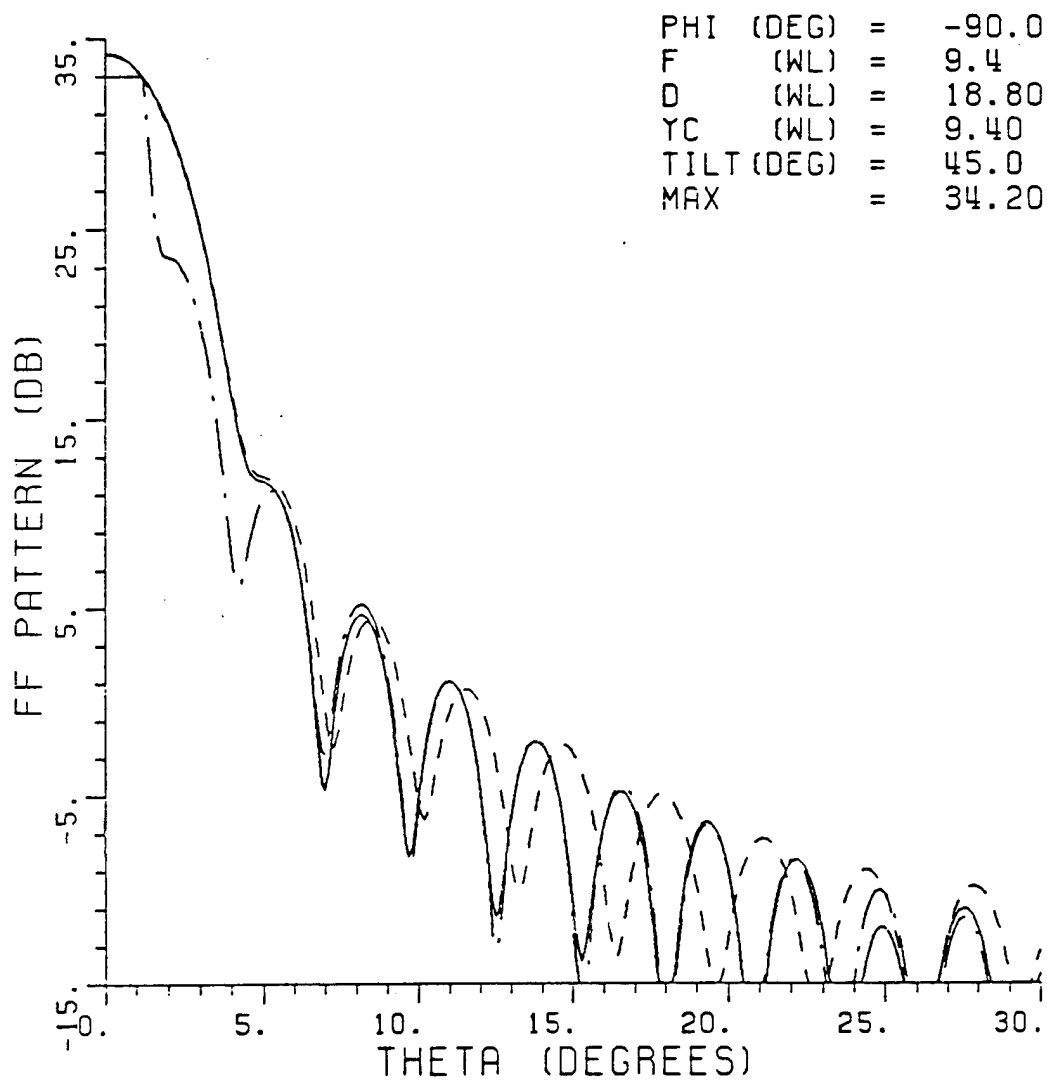
Figure 2. Far field patterns of the offset reflector.

——————— AIE

- - - - - - AIC

— - — - Multi-point GTD

# TABLE 1

## INPUT DATA FOR THE PATTERN CALCULATION BY AIE

```
CM:      ***** CHUAIE.DAT *****
CM:   EXTENDED APERTURE INTEGRATION
CM: FOR AN OFFSET CIRCULAR REFLECTOR
CE:
DG:
1
3  6. 0.3  0.3  12.  0
TO: AIE ONLY
F  90.  5.
F   0   0   0   0
F   F   F   F   0
T.  T   F   0.8
F   F   F   F   F   F   0.   0.
F   F   0.  0.
AD:   LAD=T
T F
7.
4
-11. -6.
 11. -6.
 11. 16.
-11. 16.
FD:
0   T
T   0   T   1   90.  0.  F
2   0. 90.
9
  0.   0.   0.   -300.  0.
  5.  -0.1  0.   -300.  0.
 10.  -0.5  0.   -300.  0.
 15.  -1.2  0.   -300.  0.
 20.  -2.1  0.   -300.  0.
 30.  -4.6  0.   -300.  0.
 35.  -6.35 0.   -300.  0.
 45. -10.15 0.   -300.  0.
 60. -17.0  0.   -300.  0.
  0.   0.   0.   -300.  0.
  5.  -0.1  0.   -300.  0.
 10.  -0.5  0.   -300.  0.
 15.  -1.2  0.   -300.  0.
 20.  -2.1  0.   -300.  0.
 30.  -4.6  0.   -300.  0.
 35.  -6.35 0.   -300.  0.
 45. -10.15 0.   -300.  0.
 60. -17.0  0.   -300.  0.
FQ:
1  18.5
TL:
45.  6.
PZ:
1
-90.
0.  30.  0.1
F
PP:
1
1  1
1  2
XQ:
```

TABLE 2

INPUT DATA FOR THE PATTERN CALCULATION BY AIC

```
CM:        ***** CHUAIC.DAT *****
CM:   CONVENTIONAL APERTURE INTEGRATION
CE:     FOR AN OFFSET CIRCULAR REFLECTOR
DG:
1
3  6. 0.3  0.3  12.  0
TO: AIC ONLY
F 90.  5.
F   0   0   0   0
F   F   F   F   0
T   T   F  0.8
T.  F   F   F   F   F   0.   0.
F   F   0.  0.
FD:
0    T
T   0   T   1   90.  0.  F
2   0. 90.
9
 0.    0.    0.    -300.  0.
 5.   -0.1   0.    -300.  0.
10.   -0.5   0.    -300.  0.
15.   -1.2   0.    -300.  0.
20.   -2.1   0.    -300.  0.
30.   -4.6   0.    -300.  0.
35.   -6.35  0.    -300.  0.
45.  -10.15  0.    -300.  0.
60.  -17.0   0.    -300.  0.
 0.    0.    0.    -300.  0.
 5.   -0.1   0.    -300.  0.
10.   -0.5   0.    -300.  0.
15.   -1.2   0.    -300.  0.
20.   -2.1   0.    -300.  0.
30.   -4.6   0.    -300.  0.
35.   -6.35  0.    -300.  0.
45.  -10.15  0.    -300.  0.
60.  -17.0   0.    -300.  0.
FQ:
1  18.5
TL:
45.  6.
PZ:
1
-90.
0.  30.  0.1
F
PP:
1
1  1
1  2
XQ:
```

# TABLE 3

## INPUT DATA FOR THE PATTERN CALCULATION BY GTD

```
CM:      ***** CHUGTD.DAT *****
CM:           MULTI-POINT GTD
CE: FOR AN OFFSET CIRCULAR REFLECTOR
DG:
1
3  6.  0.3  0.3  12.  0
TO: GTD ONLY
F  90.  5.
F   0   0   0   0
F   F   F   F   0
T   T   F   0.8
F   F   F   F   T   F   0.   0.
T   F   0.  0.
FD:
0   T
T   0   T   1   90.  0.  F
2   0.  90.
9
 0.    0.    0.    -300.  0.
 5.   -0.1   0.    -300.  0.
10.   -0.5   0.    -300.  0.
15.   -1.2   0.    -300.  0.
20.   -2.1   0.    -300.  0.
30.   -4.6   0.    -300.  0.
35.   -6.35  0.    -300.  0.
45.  -10.15  0.    -300.  0.
60.  -17.0   0.    -300.  0.
 0.    0.    0.    -300.  0.
 5.   -0.1   0.    -300.  0.
10.   -0.5   0.    -300.  0.
15.   -1.2   0.    -300.  0.
20.   -2.1   0.    -300.  0.
30.   -4.6   0.    -300.  0.
35.   -6.35  0.    -300.  0.
45.  -10.15  0.    -300.  0.
60.  -17.0   0.    -300.  0.
FQ:
1  18.5
TL:
45.  6.
PZ:
1
-90.
0.  30.  0.1
F
PP:
1
1  1
1  2
XQ:
```

Example 2:

This example illustrates the feed/plate scattering model in which
the forward scattered fields of a flat plate can be computed by the
Extended Aperture Integration. The plate must be perpendicular to the
axis of reflector. The default reflector with a square (2.4" x 2.4")
feed blockage model is used since the scattering fields of the feed
blockage can also be obtained by using the FB: command. The patterns
obtained from the FB: command are given in Figure 3 for the total,
reflector and feed scattering fields. The scattering fields of the feed
blockage are computed by the physical optics solution for the scattering
of a flat plate. Next the AD: command is used to calculate the
scattered fields of the square plate. Then the total fields are
obtained by subtracting the scattering fields from the reflector fields.
This is done in a separate computer code (named FDSCAT23) which
processes the output data from the reflector code. The resultant
patterns of the total and feed scattering fields are shown in Figure 4.
As can be seen by comparing Figures 3 and 4, the two feed scattering
models are in very good agreement. Note that when the AD: command
model is used, the phase reference point must be identical for the
reflector fields and the scattered fields. The input data for the
scattered field computation by AIE are given in Table 4. The input data
for the computation by FB: Command are given in Table 5.

Note that in order to use FDSCAT23.FOR to calculate the total
field, two data files have to be generated. One is the file for the
scattered field and the other is the file for the reflector field.
These data files are generated by using the PP: Copmmand. Thus, the
format of PP: Command must be the same. In this example, the reflector
field can be calculated by using the input data given in Table 4 except
that the AD: Command has to be deleted and the grid sizes in the DG:
Command have to be changed so that the number of grids does not exceed
200.

PHI (DEG) = 0.0
F (WL) = 7.5
D (WL) = 22.35
MAX = 34.64
AIC GTD

XOO (INCHES) = 0.0     0.0     4.5

(a) total fields



PHI (DEG) = 0.0
F (WL) = 7.5
D (WL) = 22.35
MAX = 34.96
AIC GTD
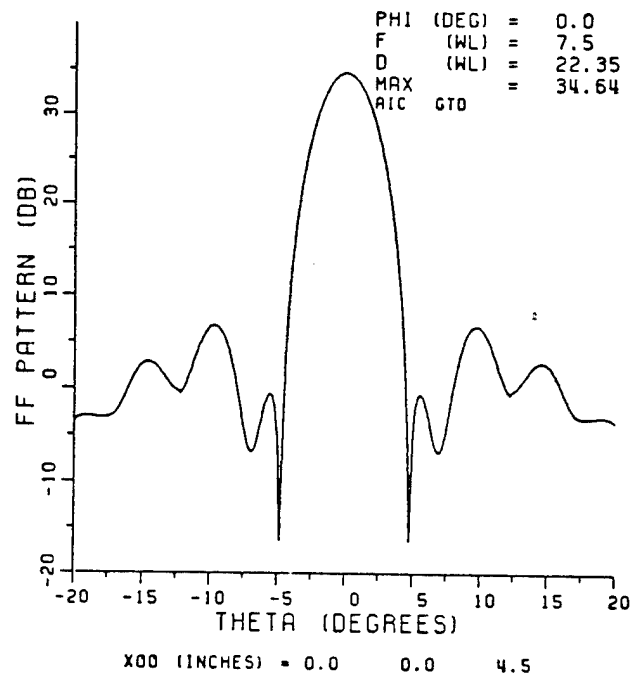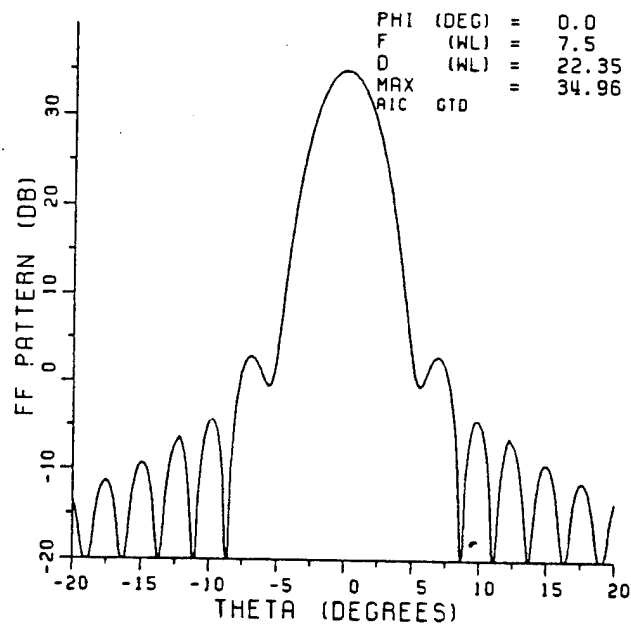
(b) reflector fields

Figure 3.  Principal patterns of the circular reflector with a square
feed blockage model.  (From FB: command).

10-10

PHI (DEG) = 0.0
F   (WL) = 7.5
D   (WL) = 22.35
MAX     = 6.11
AIC  GTD

XOO (INCHES) = 0.0    0.0    4.5
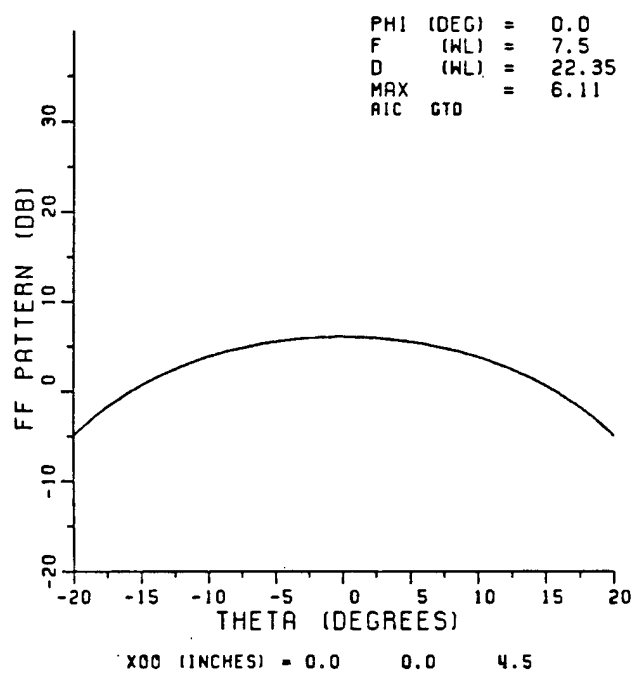
(c) feed blockage (from FB:)

Figure 3.  (Continued)

10-11
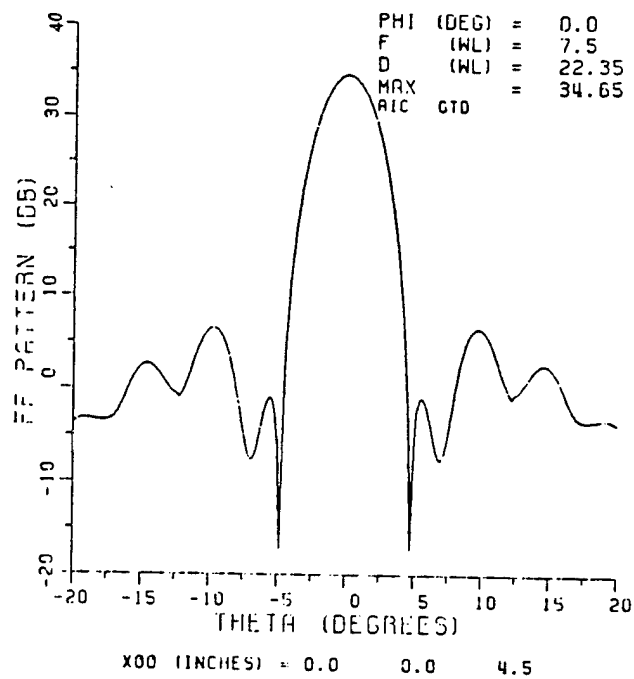
(a) total fields
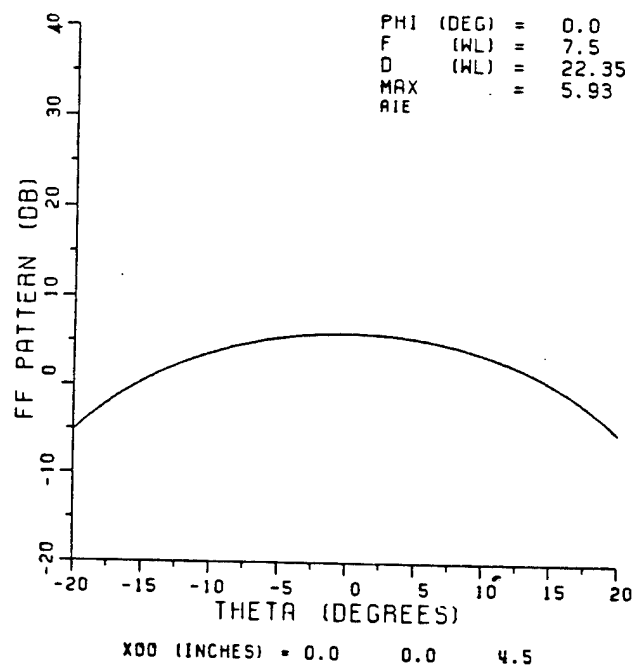


(b) feed blockage (From AD:)

Figure 4. Principal patterns of the circular reflector with a square
plate as the feed blockage model. (From AD: command)

# TABLE 4

## INPUT DATA FOR THE CALCULATION OF FEED BLOCKAGE BY AIE

```
CM:         ***** C24FBAD.DAT *****
CM: EXAMPLE OF EXTENDED APERTURE INTEGRATION
CE:      FEED BLOCKAGE MODEL  (LAD)
DG:            DEFAULT REFLECTOR
1
3   8.0  0.1  0.1  24.  0
TO: AIE ONLY
F    30.  1.
F    0    0    0    0
F    F    F    F    0
T    T    F    0.8
F :  F    F    F    F    F   0.  0.
F    F    0.  0.
AD:
T  F        DIFFRACTED FIELD INCLUDED
8.0
4
-1.2 -1.2
 1.2 -1.2
 1.2  1.2
-1.2  1.2
NF:
0.  0.  4.5    PHASE REFERNECE POINT
F
F
PZ:
1
0.
-20.  20.  0.2
F
PP:
1
1  1
1  2
XQ:
```

10-13

TABLE 5

INPUT DATA FOR THE CALCULATION OF FEED BLOCKAGE BY FB: COMMAND

```
CM:          ***** C24FB.DAT *****
CM: EXAMPLE OF EXTENDED APERTURE INTEGRATION
CE:       FEED BLOCKAGE MODEL  (FB)
DG:            DEFAULT REFLECTOR
1
3   8.0  0.6  0.6  24.  0
TO: AIE ONLY
F    30.  1.
F    0    0    0    0
F    F    F    F    0
T    T    F    0.8
T.   F    F    F    T    F    0.  0.
T    T    0.  0.
FB:
2   0.   0.   8.0                 ! RECTANGULAR FEED BLOCKAGE
2.4  2.4
NF:
0.  0.  4.5     PHASE REFERNECE POINT
F
F
PZ:
1
0.
-20.  20.  0.2
F
PP:
3
1    1
1    2
2    1
1    2
3    1
1    2
XQ:
```

Example 3:

   This example illustrates the feed/plate scattering model for an
offset pi-shaped reflector antenna.  The geometry of this antenna with
feed mast is given in Figure 5(a).  The equivalent plate scatterer is
shown in the Figure 5(b).  The FB:  Command model for feed blockage
cannot be used in this example because most of the feed and mast are
located outside the GO region of the reflector as shown in Figure 6.  In
order to calculate the field from this scatterer, the Extended Aperture
Integration has to be used.  The aperture field is calculated from the
edge diffracted field of the reflector.  The input data for calculating
the feed scatter is given in Table 6 with the AD:  Command used to
specify the plate scatterer.  The calculated E-plane (offset plane)
scattered field at 35 GHz is given in Figure 7.  The reflector pattern
without the plate is calculated by using the same data as given in Table
6 but without the AD:  Command and is shown in Figure 8.  The total
field is then calculated by subtracting the scattered field from the
reflector field (by using FDSCAT23.FOR) and is shown in Figure 9.  Note
that in the calculations, a common phase center is input in the NF:
Command.  The measured E-plane pattern is shown in Figure 10.  Comparing
Figures 8, 9 and 10, it is found the agreement between calculated and
measured patterns for $-10° < \theta < 0°$ is improved after the feed scattering
effect is included.  The differences for $0° < \theta < 10°$ are probably due to
the fact that reflections from the feed mast are not modeled in the
calculation.
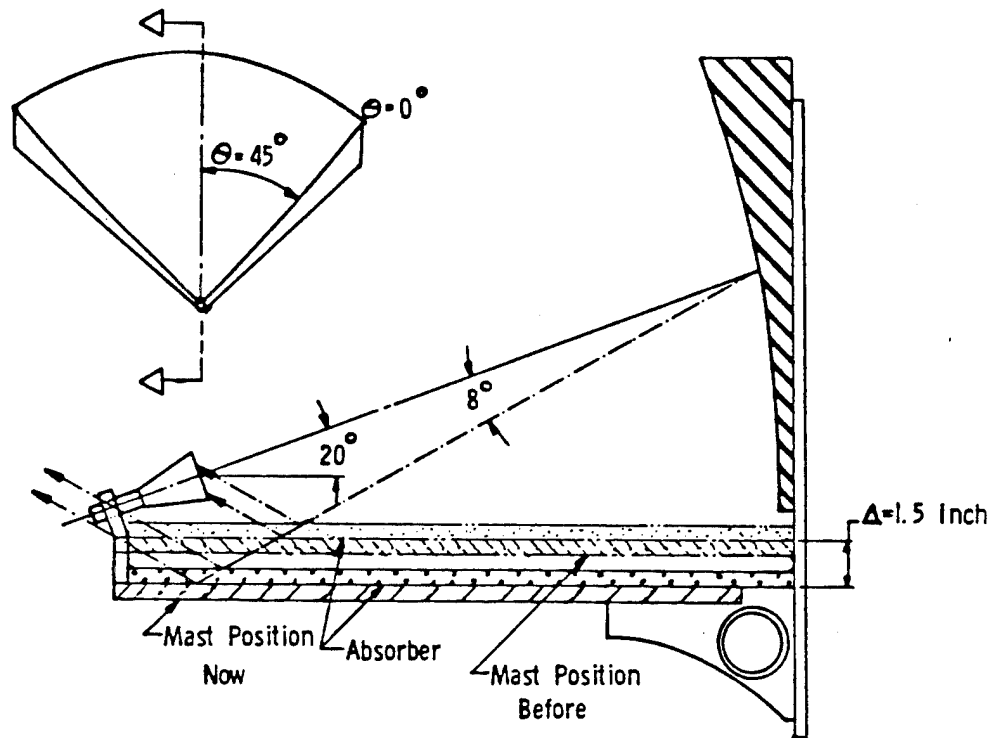
(a) Antenna with feed mast
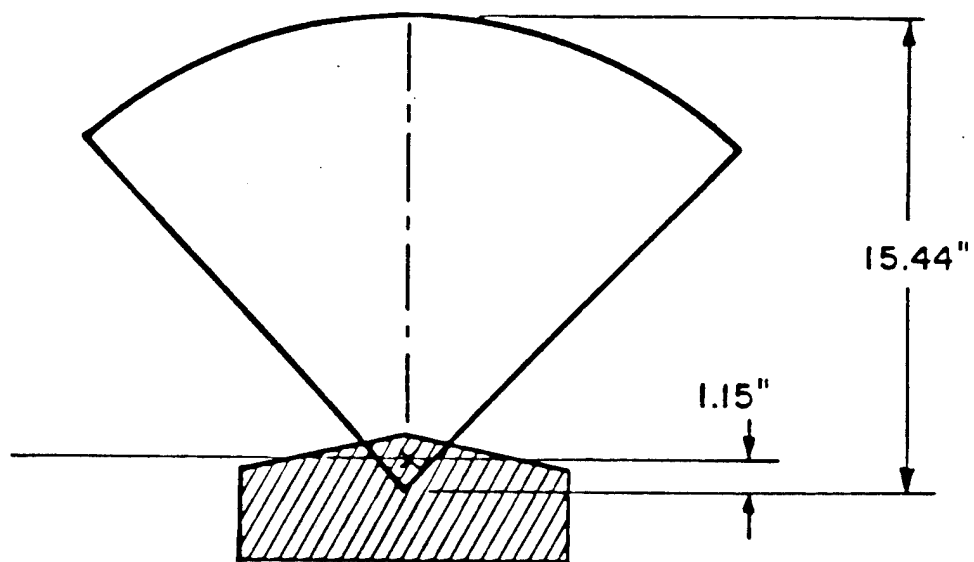


f = 35.0 GHz

F = 20.3" = 60.16 λ

(b) Equivalnet plate scatterer

Figure 5.   Equivalent plate scatterer for the feed and mast.

10-16
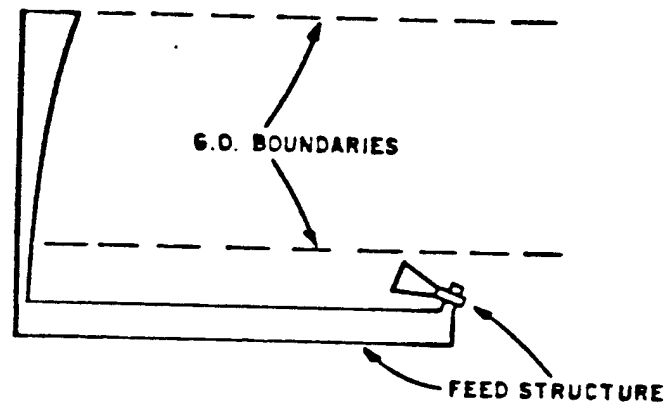
(a) Offset feed structure
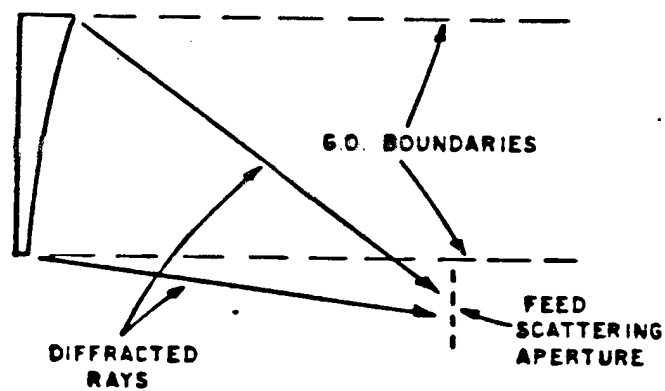


(b) Feed scattering aperture

Figure 6. Geometry of feed scattering for offset reflector.
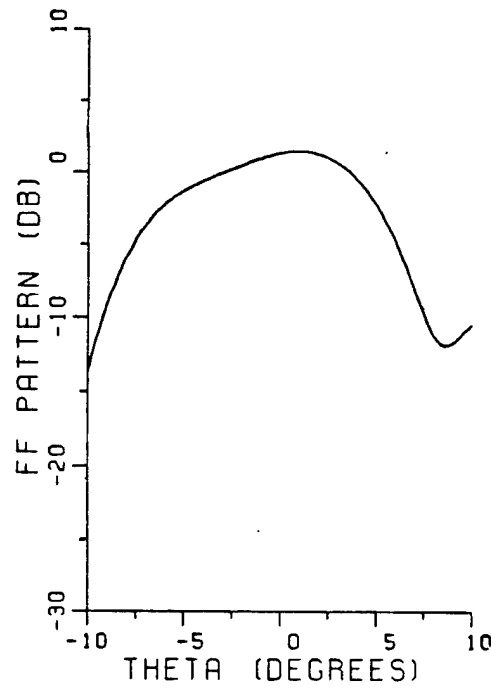
Figure 7. Scattered field from feed scatter by the AD: Command of Example 3 at 35 GHz.
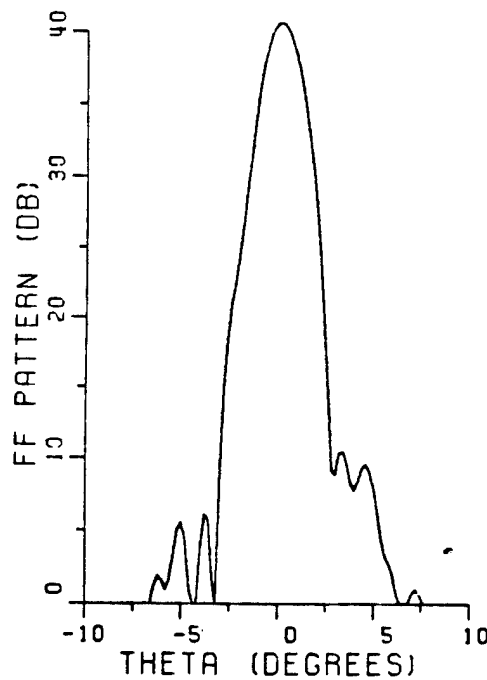


Figure 8. Reflector field for the offset reflector of Example 3 at 35 GHz.

10-18
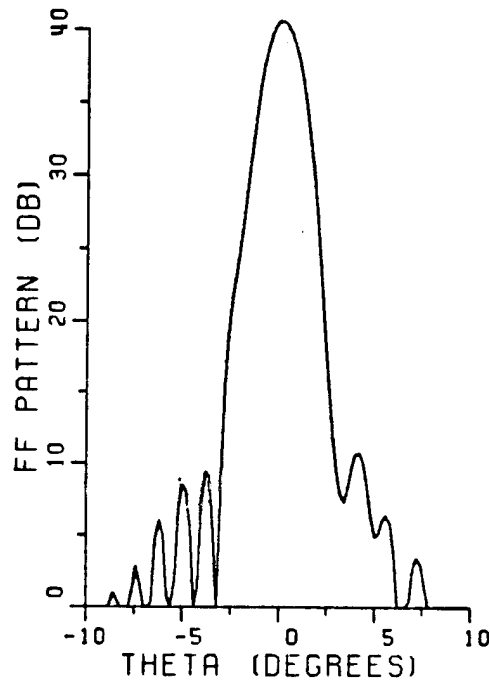
Figure 9. Calculated total field of the offset reflector at 35 GHz.
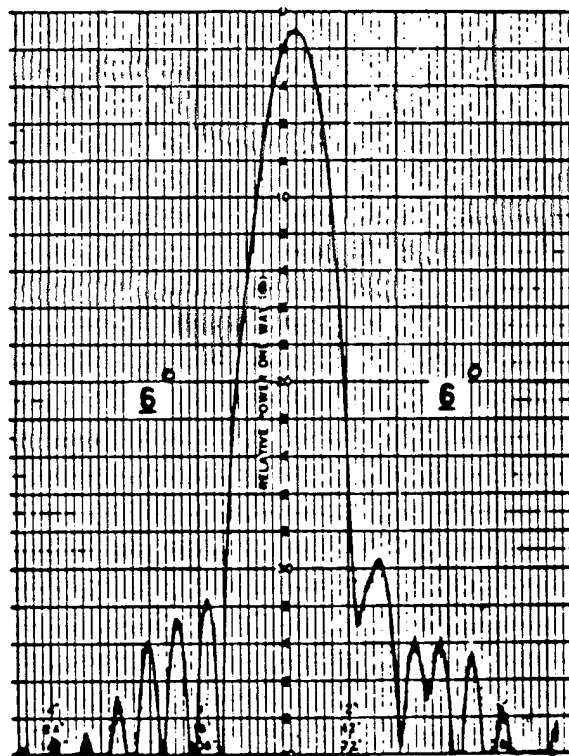


Figure 10. Measured E-plane (offset plane) pattern of the pi-shaped reflector at 35 GHz.

# TABLE 6

## INPUT DATA FOR SCATTERED FIELD CALCULATION OF EXAMPLE 3

```
CM:        ***** LARC35A.DAT *****
CM: USE AD: TO MODIFY THE FEED BLOCKAGE
CM:            FEED BLOCKAGE
CE:        PI-SHAPED REFLECTOR
DG:
1
3   20.3    0.35    0.35    0.  20
 10.66       9.51
  9.19      10.95
  8.20      11.71
  7.15      12.38
  6.04      12.95
  4.89      13.43
  3.70      13.80
  2.48      14.07
  1.25      14.24
  0.00      14.29
 -1.25      14.24
 -2.48      14.07
 -3.70      13.80
 -4.89      13.43
 -6.04      12.95
 -7.15      12.38
 -8.20      11.71
 -9.19      10.95
-10.66       9.51
  0.00      -1.15
AD:
T    F
20.3
5
-5.33     -3.00
 5.33     -3.00
 5.33     -0.22
 0.00      1.00
-5.33     -0.22
NF:
0.    6.57    2.517
F
F
TO: LAIC = T ONLY
F    0.    0.
F    0     0     0     0
F    F     F     F     0
T    T     F     0.8
T    F     F     F     F     F     0.    0.
F    F     0.    0.
FD:
0    T
T    0     T     1    '0.    0.   F
2    0.   90.
37
 0.000       10.170       55.306      -300.    0.
 2.500        9.974       55.603      -300.    0.
 5.000        9.383       56.521      -300.    0.
 7.500        8.385       58.155      -300.    0.
10.000        6.962       60.691      -300.    0.
12.500        5.085       64.461      -300.    0.
15.000        2.722       70.061      -300.    0.
17.500       -0.143       78.571      -300.    0.
20.000       -3.431       91.869      -300.    0.
22.500       -6.730      112.309      -300.    0.
25.000       -9.125      138.960      -300.    0.
27.500      -10.240      164.140      -300.    0.
```

TABLE 6 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 30.000 | -10.944 | -177.104 | -300. | 0. |
| 32.500 | -12.007 | -163.192 | -300. | 0. |
| 35.000 | -13.714 | -151.218 | -300. | 0. |
| 37.500 | -16.149 | -138.470 | -300. | 0. |
| 40.000 | -19.267 | -121.546 | -300. | 0. |
| 42.500 | -22.539 | -95.571 | -300. | 0. |
| 45.000 | -24.314 | -60.728 | -300. | 0. |
| 47.500 | -24.028 | -31.179 | -300. | 0. |
| 50.000 | -23.365 | -12.571 | -300. | 0. |
| 52.500 | -23.178 | -0.557 | -300. | 0. |
| 55.000 | -23.563 | 8.303 | -300. | 0. |
| 57.500 | -24.455 | 15.829 | -300. | 0. |
| 60.000 | -25.785 | 23.110 | -300. | 0. |
| 62.500 | -27.488 | 31.013 | -300. | 0. |
| 65.000 | -29.489 | 40.446 | -300. | 0. |
| 67.500 | -31.653 | 52.463 | -300. | 0. |
| 70.000 | -33.711 | 67.960 | -300. | 0. |
| 72.500 | -35.241 | 86.490 | -300. | 0. |
| 75.000 | -35.945 | 105.186 | -300. | 0. |
| 77.500 | -35.977 | 120.718 | -300. | 0. |
| 80.000 | -35.724 | 131.981 | -300. | 0. |
| 82.500 | -35.459 | 139.564 | -300. | 0. |
| 85.000 | -35.296 | 144.353 | -300. | 0. |
| 87.500 | -35.270 | 146.990 | -300. | 0. |
| 90.000 | -35.388 | 147.833 | -300. | 0. |
| 0.000 | 10.130 | 55.523 | -300. | 0. |
| 2.500 | 9.838 | 56.168 | -300. | 0. |
| 5.000 | 8.948 | 58.232 | -300. | 0. |
| 7.500 | 7.430 | 62.170 | -300. | 0. |
| 10.000 | 5.243 | 68.999 | -300. | 0. |
| 12.500 | 2.405 | 80.817 | -300. | 0. |
| 15.000 | -0.718 | 101.166 | -300. | 0. |
| 17.500 | -2.886 | 130.712 | -300. | 0. |
| 20.000 | -3.319 | 158.777 | -300. | 0. |
| 22.500 | -3.311 | 177.876 | -300. | 0. |
| 25.000 | -3.924 | -169.135 | -300. | 0. |
| 27.500 | -5.495 | -158.320 | -300. | 0. |
| 30.000 | -8.202 | -146.131 | -300. | 0. |
| 32.500 | -12.191 | -126.457 | -300. | 0. |
| 35.000 | -16.183 | -85.910 | -300. | 0. |
| 37.500 | -15.580 | -37.846 | -300. | 0. |
| 40.000 | -13.416 | -13.549 | -300. | 0. |
| 42.500 | -12.388 | -1.306 | -300. | 0. |
| 45.000 | -12.484 | 6.765 | -300. | 0. |
| 47.500 | -13.541 | 13.624 | -300. | 0. |
| 50.000 | -15.526 | 21.090 | -300. | 0. |
| 52.500 | -18.541 | 31.666 | -300. | 0. |
| 55.000 | -22.682 | 51.670 | -300. | 0. |
| 57.500 | -26.146 | 94.642 | -300. | 0. |
| 60.000 | -24.543 | 139.284 | -300. | 0. |
| 62.500 | -21.824 | 159.975 | -300. | 0. |
| 65.000 | -20.013 | 169.901 | -300. | 0. |
| 67.500 | -19.033 | 175.716 | -300. | 0. |
| 70.000 | -18.539 | 179.637 | -300. | 0. |
| 72.500 | -18.397 | -177.467 | -300. | 0. |
| 75.000 | -18.490 | -175.205 | -300. | 0. |
| 77.500 | -18.732 | -173.387 | -300. | 0. |
| 80.000 | -19.056 | -171.926 | -300. | 0. |
| 82.500 | -19.408 | -170.784 | -300. | 0. |
| 85.000 | -19.745 | -169.957 | -300. | 0. |
| 87.500 | -20.036 | -169.453 | -300. | 0. |
| 90.000 | -20.261 | -169.284 | -300. | 0. |

FQ:
1    35.
TL:
20.0    0.

TABLE 6 - CONTINUED

```
PZ:
1
-90.
-10.   10.   0.2
F
PP:
1
1   1
1   2
XQ:
```

------------------------------------------------------
COMMAND AP:   INPUT APERTURE FIELD            -
------------------------------------------------------


     This command enables the user to directly input the aperture field
distribution without using a feed pattern (FD: Command).  The aperture
shape is controlled by the DG: Command.  Several types of magnitude
distributions and phase distributions are available to use.  See tables
below.  If necessary, other distributions can be added as desired in
subroutine EAP.  Note that if AP:  command is used, the input from the
FD:  Command will be ignored.




                BLOCK DIAGRAM FOR APERTURE FIELD INPUT




                        |COMMAND AP:|
                             |
                             |
                             |
             |READ:   NMAG,(CC(N),N=1,6)|
                             |
                             |
                             |
             |READ:   NPHA,(PP(N),N=1,6)|
                             |
                             |
                             |
                     |READ:  TAUP|
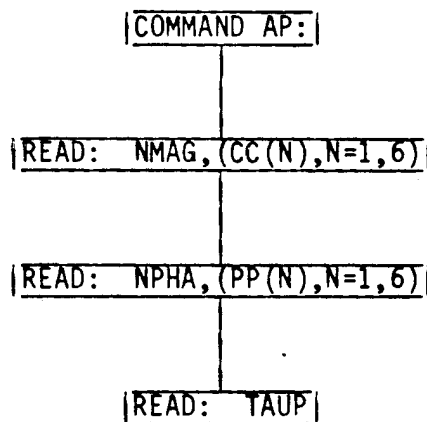
1. READ: NMAG,(CC(N),N=1,6)

    a) NMAG: This integer variable specifies which type of magnitude distribution is to be used. Several types of magnitude distribution are available as shown below:

| NMAG | MAGNITUDE DISTRIBUTION |
|------|------------------------|
| 1 | Uniform magnitude |
| 2 | Linear variation in X direction: $CC(1)$ dB taper. |
| 3 | Linear variation in Y direction: $CC(1)$ dB taper. |
| 4 | Linear variation in RHO direction: $CC(1)$ dB taper. (Circular aperture only) |
| 5 | Quadratic in X direction: $1+CC(1)X_n+CC(2)X_n^2$ |
| 6 | Quadratic in Y direction: $1+CC(1)Y_n+CC(2)Y_n^2$ |
| 7 | Quadratic in RHO direct.: $1+CC(1)R_n+CC(2)R_n^2$ (Circular aperture only) |
| 10 | Rectangular wave guide $TE_{10}$ mode |
| 11 | Circular wave guide $TE_{11}$ mode |
| 12 | Circular wave guide $TM_{11}$ mode |
| 13 | Circular wave guide $TE_{11}$ and $TM_{11}$ mode. The field is defined as $E_{TM11}/E_{TE11}=CC(1)e^{jCC(2)}$ |
| 14 | Circular waveguide $HE_{11}$ mode. |
| 20 | see below |

The aperture fields of circular waveguide $TE_{11}$ and $TM_{11}$ modes have been normalized so that the power carried by each mode is unity.

For NMAG=20, the aperture field in dB is defined as follows:

$$EAP_{dB} = - a_1 \left( \frac{\rho}{\rho_1} \right)^2 \qquad\qquad 0 < \rho < \rho_1$$

$$EAP_{dB} = - a_1 \left( \frac{\rho}{\rho_1} \right)^2 - \left( a_2 - \frac{a_1}{\rho_1^2} \right) \left( \frac{\rho - \rho_1}{1 - \rho_1} \right)^2 \qquad \rho_1 < \rho < 1$$

where $\rho$ is the normalized radial distance from aperture center, and $a_1$ and $a_2$ specify the aperture field in dB at $\rho_1$ and 1, respectively. Thus, the aperture field is $-a_1$ dB at $\rho = \rho_1$ and is $-a_2$ dB at $\rho = 1$. $a_1$, $\rho_1$ and $a_2$ are input in CC(1), CC(2), and CC(3), respectively.

b) CC(N): This dimensioned real variable is used to supply the necessary constants for the magnitude distribution. NOTE: $X_n$, $Y_n$, and $R_n$ in the above table are normalized quantities as measured from the center (Xo,Yo) of the aperture where Xo=0.5 (Xmax+Xmin)and Yo=0.5 (Ymax+Ymin). For example, NMAG=5, CC(1)=0 and CC(2)=-0.8 gives an aperture distribution that has a -14 dB edge illumination along the X direction. A circular aperture that goes to zero at the rim can be obtained by using D>0 in the DG: Command and NMAG=7, CC(1)=0 and CC(2)=-1. When MNAG=13, CC(1) and CC(2) define the field ratio between $TM_{11}$ and $TE_{11}$ modes as shown in the table where CC(1) is the magnitude and CC(2) is the phase shift in degrees.

When NMAG=11 and the radius of the waveguide is smaller than the cutoff radius of the $TE_{11}$ mode, CC(1)=1 specifies that normalization constant of the waveguide mode is ignored so that one can still use AP: Command to simulate elements of microstrip antenna.

Currently, only the first two constants are used at most, but an array dimension of 6 is reserved for future use. Note that 6 values must be input the CC(N), although only one or two are used.

2. READ: NPHA,(PP(N),N=1,6)

    a) NPHA: This integer variable specifies the type of phase distribution to be used. Several types of phase distribution are available as shown below:

| NPHA | PHASE DISTRIBUTION |
|------|--------------------|
| 1 | Constant phase of PP(1) degrees. |
| 2 | Phase for beam scanned PP(1) degrees in X direction. |
| 3 | Phase for beam scanned PP(1) degrees in Y direction. |
| 4 | Phase for beam scanned PP(1) degrees in the Theta direction (from Z-axis) and PP(2) degrees in the PHI direction (from XY plane). |
| 10 | Rectangular horn phase with horn flare angles: PP(1), PP(2) degrees in X and Y, respectively. |
| 11 | Conical horn phase with horn flare angle: PP(1) degrees. |

    b) PP(N): This dimensioned real variable is used to supply the necessary constants for phase distribution. Similar to CC(N), at most 2 out of 6 are currently used.

3. READ: TAUP

    a) TAUP: This real variable is input in degrees and defines the linear polarization angle of the aperture field.

Example 1:   This example illustrates the use of the AP: Command to calculate the radiation patterns of a conical horn antenna. This horn antenna has diameter of 1.2" and flare angle of 14.9 degrees and is operated at 38 GHz. The input data are given in Table 1 and the patterns of this horn are given in Figure 1. Note also that LFDOUT in the PZ: Command is true so the output patterns are stored in Unit #7 which is given in Table 2, and can be used as linear feed input in the FD: Command of other runs such as the examples in the DG: Command.

# TABLE 1

## INPUT DATA FOR CONICAL HORN PATTERN CALCULATION

```
CM:              ***** CONI.DAT *****
CM:    USE AP: TO GENERATE CONICAL HORN PATTERNS
CM: CONICAL HORN : 1.2" DIAMETER, 14.9 DEG. FLARE
CE:                 38.0 GHZ
DG:
1
3    2.   0.02   0.02   1.2    0
AP:
11    0.    0.   0.   0.   0.   0.
11   14.9   0.   0.   0.   0.   0.
90.
FQ:
1    38.
PZ:
3
0.   45.   90.
0.   90.   1.
T
PP:
1
1   1
1   2
XQ:
```

# TABLE 2

## OUTPUT DATA FROM UNIT #7 OF THE CONICAL HORN PATTERNS

| | | | | |
|---|---|---|---|---|
| 0.000 | 35.510 | 71.481 | -47.871 | -110.173 |
| 1.000 | 35.478 | 71.505 | -47.992 | -111.291 |
| 2.000 | 35.383 | 71.578 | -48.288 | -112.285 |
| 3.000 | 35.223 | 71.701 | -48.770 | -113.173 |
| 4.000 | 34.999 | 71.875 | -49.454 | -113.969 |
| 5.000 | 34.710 | 72.104 | -50.369 | -114.681 |
| 6.000 | 34.354 | 72.392 | -51.558 | -115.314 |
| 7.000 | 33.929 | 72.744 | -53.093 | -115.860 |
| 8.000 | 33.435 | 73.165 | -55.099 | -116.303 |
| 9.000 | 32.869 | 73.665 | -57.822 | -116.590 |
| 10.000 | 32.229 | 74.255 | -61.850 | -116.530 |
| 11.000 | 31.510 | 74.947 | -69.377 | -114.909 |
| 12.000 | 30.710 | 75.760 | -79.011 | 50.127 |
| 13.000 | 29.824 | 76.716 | -65.522 | 58.389 |
| 14.000 | 28.845 | 77.846 | -60.826 | 59.005 |
| 15.000 | 27.768 | 79.191 | -58.113 | 59.058 |
| 16.000 | 26.584 | 80.805 | -56.342 | 58.974 |
| 17.000 | 25.283 | 82.765 | -55.146 | 58.856 |
| 18.000 | 23.853 | 85.183 | -54.355 | 58.748 |
| 19.000 | 22.282 | 88.218 | -53.878 | 58.670 |
| 20.000 | 20.555 | 92.112 | -53.661 | 58.641 |
| 21.000 | 18.662 | 97.235 | -53.672 | 58.679 |
| 22.000 | 16.609 | 104.154 | -53.894 | 58.805 |
| 23.000 | 14.446 | 113.694 | -54.318 | 59.046 |
| 24.000 | 12.330 | 126.825 | -54.945 | 59.441 |
| 25.000 | 10.596 | 143.883 | -55.782 | 60.050 |
| 26.000 | 9.629 | 163.071 | -56.848 | 60.962 |
| 27.000 | 9.473 | -179.206 | -58.172 | 62.328 |
| 28.000 | 9.766 | -165.194 | -59.802 | 64.410 |
| 29.000 | 10.142 | -154.821 | -61.812 | 67.710 |
| 30.000 | 10.411 | -147.137 | -64.313 | 73.294 |
| 31.000 | 10.504 | -141.259 | -67.429 | 83.703 |
| 32.000 | 10.406 | -136.563 | -70.952 | 105.147 |
| 33.000 | 10.117 | -132.626 | -72.735 | 143.072 |
| 34.000 | 9.643 | -129.158 | -70.954 | 175.949 |
| 35.000 | 8.988 | -125.941 | -68.502 | -167.552 |
| 36.000 | 8.150 | -122.795 | -66.632 | -159.442 |
| 37.000 | 7.125 | -119.542 | -65.340 | -155.083 |
| 38.000 | 5.899 | -115.981 | -64.498 | -152.624 |
| 39.000 | 4.455 | -111.840 | -64.010 | -151.273 |
| 40.000 | 2.766 | -106.706 | -63.813 | -150.652 |
| 41.000 | 0.809 | -99.895 | -63.869 | -150.579 |
| 42.000 | -1.414 | -90.215 | -64.154 | -150.973 |
| 43.000 | -3.774 | -75.700 | -64.660 | -151.822 |
| 44.000 | -5.757 | -54.353 | -65.386 | -153.173 |
| 45.000 | -6.452 | -28.471 | -66.343 | -155.146 |
| 46.000 | -5.730 | -5.947 | -67.552 | -157.969 |
| 47.000 | -4.471 | 9.599 | -69.045 | -162.059 |
| 48.000 | -3.281 | 19.812 | -70.859 | -168.201 |
| 49.000 | -2.333 | 26.781 | -73.001 | -177.903 |
| 50.000 | -1.637 | 31.815 | -75.292 | 166.150 |
| 51.000 | -1.162 | 35.652 | -76.946 | 141.521 |
| 52.000 | -0.878 | 38.717 | -76.788 | 113.449 |
| 53.000 | -0.756 | 41.267 | -75.205 | 92.215 |
| 54.000 | -0.775 | 43.467 | -73.386 | 79.027 |
| 55.000 | -0.919 | 45.425 | -71.810 | 70.819 |
| 56.000 | -1.175 | 47.221 | -70.540 | 65.393 |
| 57.000 | -1.533 | 48.913 | -69.539 | 61.577 |
| 58.000 | -1.986 | 50.549 | -68.760 | 58.748 |
| 59.000 | -2.527 | 52.170 | -68.164 | 56.560 |
| 60.000 | -3.154 | 53.814 | -67.719 | 54.806 |

11-7

TABLE 2 - CONTINUED

| | | | |
|---|---|---|---|
| 61.000 | -3.864 | 55.521 | -67.402 | 53.360 |
| 62.000 | -4.656 | 57.331 | -67.193 | 52.138 |
| 63.000 | -5.530 | 59.293 | -67.077 | 51.083 |
| 64.000 | -6.486 | 61.466 | -67.043 | 50.156 |
| 65.000 | -7.527 | 63.924 | -67.081 | 49.330 |
| 66.000 | -8.654 | 66.765 | -67.181 | 48.582 |
| 67.000 | -9.870 | 70.116 | -67.338 | 47.898 |
| 68.000 | -11.172 | 74.154 | -67.543 | 47.266 |
| 69.000 | -12.556 | 79.116 | -67.793 | 46.677 |
| 70.000 | -14.002 | 85.311 | -68.081 | 46.123 |
| 71.000 | -15.467 | 93.116 | -68.403 | 45.600 |
| 72.000 | -16.864 | 102.886 | -68.754 | 45.102 |
| 73.000 | -18.053 | 114.718 | -69.130 | 44.627 |
| 74.000 | -18.870 | 128.088 | -69.526 | 44.171 |
| 75.000 | -19.213 | 141.735 | -69.940 | 43.734 |
| 76.000 | -19.126 | 154.224 | -70.365 | 43.314 |
| 77.000 | -18.760 | 164.709 | -70.798 | 42.911 |
| 78.000 | -18.269 | 173.073 | -71.235 | 42.524 |
| 79.000 | -17.759 | 179.604 | -71.672 | 42.156 |
| 80.000 | -17.286 | -175.316 | -72.104 | 41.806 |
| 81.000 | -16.876 | -171.351 | -72.526 | 41.477 |
| 82.000 | -16.537 | -168.239 | -72.935 | 41.172 |
| 83.000 | -16.268 | -165.788 | -73.326 | 40.892 |
| 84.000 | -16.067 | -163.860 | -73.695 | 40.640 |
| 85.000 | -15.929 | -162.354 | -74.038 | 40.420 |
| 86.000 | -15.850 | -161.200 | -74.352 | 40.235 |
| 87.000 | -15.827 | -160.348 | -74.633 | 40.088 |
| 88.000 | -15.855 | -159.761 | -74.879 | 39.981 |
| 89.000 | -15.934 | -159.418 | -75.088 | 39.916 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 35.507 | 71.502 | -36.325 | 40.356 |
| 1.000 | 35.467 | 71.541 | -22.513 | 53.085 |
| 2.000 | 35.346 | 71.659 | -11.934 | 55.588 |
| 3.000 | 35.143 | 71.858 | -5.298 | 56.211 |
| 4.000 | 34.858 | 72.144 | -0.580 | 56.482 |
| 5.000 | 34.488 | 72.522 | 3.020 | 56.656 |
| 6.000 | 34.032 | 73.004 | 5.879 | 56.795 |
| 7.000 | 33.486 | 73.601 | 8.206 | 56.925 |
| 8.000 | 32.847 | 74.333 | 10.126 | 57.058 |
| 9.000 | 32.110 | 75.222 | 11.721 | 57.200 |
| 10.000 | 31.270 | 76.300 | 13.047 | 57.357 |
| 11.000 | 30.319 | 77.610 | 14.143 | 57.531 |
| 12.000 | 29.249 | 79.210 | 15.037 | 57.726 |
| 13.000 | 28.050 | 81.185 | 15.751 | 57.945 |
| 14.000 | 26.711 | 83.653 | 16.300 | 58.192 |
| 15.000 | 25.216 | 86.791 | 16.696 | 58.472 |
| 16.000 | 23.554 | 90.862 | 16.948 | 58.790 |
| 17.000 | 21.716 | 96.272 | 17.060 | 59.153 |
| 18.000 | 19.716 | 103.636 | 17.037 | 59.569 |
| 19.000 | 17.619 | 113.819 | 16.880 | 60.050 |
| 20.000 | 15.619 | 127.724 | 16.588 | 60.611 |
| 21.000 | 14.075 | 145.315 | 16.157 | 61.271 |
| 22.000 | 13.322 | 164.272 | 15.582 | 62.058 |
| 23.000 | 13.293 | -178.889 | 14.852 | 63.010 |
| 24.000 | 13.600 | -165.822 | 13.953 | 64.185 |
| 25.000 | 13.917 | -156.151 | 12.866 | 65.669 |
| 26.000 | 14.084 | -148.917 | 11.559 | 67.602 |
| 27.000 | 14.045 | -143.297 | 9.991 | 70.222 |
| 28.000 | 13.784 | -138.708 | 8.099 | 73.965 |
| 29.000 | 13.296 | -134.746 | 5.792 | 79.707 |
| 30.000 | 12.578 | -131.107 | 2.972 | 89.397 |
| 31.000 | 11.617 | -127.528 | -0.270 | 107.537 |
| 32.000 | 10.394 | -123.735 | -2.622 | 140.162 |
| 33.000 | 8.875 | -119.365 | -1.819 | 175.584 |
| 34.000 | 7.007 | -113.836 | 0.482 | -163.245 |
| 35.000 | 4.722 | -106.068 | 2.492 | -152.051 |

TABLE 2 - CONTINUED

| | | | |
|---|---|---|---|
| 36.000 | 1.989 | -93.822 | 3.975 | -145.513 |
| 37.000 | -0.894 | -72.747 | 5.018 | -141.256 |
| 38.000 | -2.467 | -40.362 | 5.716 | -138.231 |
| 39.000 | -1.475 | -9.630 | 6.137 | -135.923 |
| 40.000 | 0.472 | 9.305 | 6.325 | -134.055 |
| 41.000 | 2.168 | 20.221 | 6.312 | -132.460 |
| 42.000 | 3.425 | 27.061 | 6.115 | -131.029 |
| 43.000 | 4.297 | 31.765 | 5.746 | -129.682 |
| 44.000 | 4.855 | 35.257 | 5.210 | -128.355 |
| 45.000 | 5.155 | 38.018 | 4.506 | -126.982 |
| 46.000 | 5.234 | 40.321 | 3.624 | -125.493 |
| 47.000 | 5.120 | 42.333 | 2.550 | -123.795 |
| 48.000 | 4.830 | 44.171 | 1.255 | -121.751 |
| 49.000 | 4.372 | 45.921 | -0.301 | -119.137 |
| 50.000 | 3.752 | 47.661 | -2.185 | -115.548 |
| 51.000 | 2.967 | 49.465 | -4.499 | -110.158 |
| 52.000 | 2.009 | 51.423 | -7.384 | -101.055 |
| 53.000 | 0.862 | 53.654 | -10.859 | -83.333 |
| 54.000 | -0.499 | 56.332 | -13.583 | -48.296 |
| 55.000 | -2.113 | 59.746 | -12.465 | -8.751 |
| 56.000 | -4.037 | 64.413 | -9.642 | 13.079 |
| 57.000 | -6.344 | 71.360 | -7.266 | 23.881 |
| 58.000 | -9.085 | 82.782 | -5.484 | 29.978 |
| 59.000 | -11.995 | 103.163 | -4.149 | 33.857 |
| 60.000 | -13.591 | 135.658 | -3.140 | 36.550 |
| 61.000 | -12.441 | 166.940 | -2.377 | 38.544 |
| 62.000 | -10.236 | -174.167 | -1.805 | 40.093 |
| 63.000 | -8.264 | -163.596 | -1.384 | 41.343 |
| 64.000 | -6.708 | -157.174 | -1.088 | 42.382 |
| 65.000 | -5.503 | -152.912 | -0.895 | 43.268 |
| 66.000 | -4.569 | -149.877 | -0.790 | 44.040 |
| 67.000 | -3.845 | -147.597 | -0.758 | 44.725 |
| 68.000 | -3.286 | -145.813 | -0.791 | 45.341 |
| 69.000 | -2.860 | -144.371 | -0.879 | 45.903 |
| 70.000 | -2.541 | -143.177 | -1.014 | 46.422 |
| 71.000 | -2.312 | -142.166 | -1.190 | 46.905 |
| 72.000 | -2.157 | -141.297 | -1.401 | 47.359 |
| 73.000 | -2.064 | -140.540 | -1.642 | 47.787 |
| 74.000 | -2.024 | -139.874 | -1.908 | 48.194 |
| 75.000 | -2.028 | -139.282 | -2.193 | 48.581 |
| 76.000 | -2.069 | -138.754 | -2.495 | 48.951 |
| 77.000 | -2.141 | -138.281 | -2.808 | 49.303 |
| 78.000 | -2.238 | -137.856 | -3.129 | 49.638 |
| 79.000 | -2.356 | -137.474 | -3.453 | 49.956 |
| 80.000 | -2.490 | -137.132 | -3.777 | 50.255 |
| 81.000 | -2.637 | -136.827 | -4.098 | 50.535 |
| 82.000 | -2.794 | -136.557 | -4.411 | 50.794 |
| 83.000 | -2.956 | -136.320 | -4.714 | 51.029 |
| 84.000 | -3.123 | -136.116 | -5.004 | 51.239 |
| 85.000 | -3.291 | -135.944 | -5.277 | 51.422 |
| 86.000 | -3.459 | -135.804 | -5.532 | 51.575 |
| 87.000 | -3.625 | -135.694 | -5.765 | 51.696 |
| 88.000 | -3.787 | -135.616 | -5.977 | 51.784 |
| 89.000 | -3.946 | -135.570 | -6.165 | 51.837 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 35.505 | 71.492 | -111.111 | 90.176 |
| 1.000 | 35.457 | 71.546 | -64.215 | -25.526 |
| 2.000 | 35.310 | 71.710 | -58.300 | -25.574 |
| 3.000 | 35.063 | 71.990 | -54.967 | -25.497 |
| 4.000 | 34.716 | 72.395 | -52.736 | -25.350 |
| 5.000 | 34.263 | 72.939 | -51.145 | -25.141 |
| 6.000 | 33.700 | 73.645 | -49.991 | -24.870 |
| 7.000 | 33.022 | 74.541 | -49.167 | -24.529 |
| 8.000 | 32.221 | 75.670 | -48.609 | -24.111 |
| 9.000 | 31.287 | 77.092 | -48.281 | -23.603 |
| 10.000 | 30.207 | 78.893 | -48.159 | -22.990 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 11.000 | 28.967 | 81.203 | -48.229 | -22.248 |
| 12.000 | 27.547 | 84.218 | -48.485 | -21.346 |
| 13.000 | 25.927 | 88.251 | -48.926 | -20.241 |
| 14.000 | 24.089 | 93.808 | -49.558 | -18.868 |
| 15.000 | 22.039 | 101.720 | -50.393 | -17.133 |
| 16.000 | 19.862 | 113.259 | -51.450 | -14.889 |
| 17.000 | 17.851 | 129.802 | -52.757 | -11.893 |
| 18.000 | 16.579 | 150.883 | -54.352 | -7.732 |
| 19.000 | 16.425 | 172.085 | -56.280 | -1.646 |
| 20.000 | 17.022 | -171.094 | -58.563 | 7.815 |
| 21.000 | 17.785 | -159.214 | -61.059 | 23.310 |
| 22.000 | 18.409 | -150.905 | -63.042 | 47.658 |
| 23.000 | 18.797 | -144.871 | -63.260 | 76.455 |
| 24.000 | 18.933 | -140.255 | -61.956 | 98.783 |
| 25.000 | 18.825 | -136.520 | -60.388 | 112.718 |
| 26.000 | 18.480 | -133.315 | -59.103 | 121.421 |
| 27.000 | 17.900 | -130.390 | -58.182 | 127.249 |
| 28.000 | 17.080 | -127.538 | -57.594 | 131.460 |
| 29.000 | 16.001 | -124.550 | -57.300 | 134.724 |
| 30.000 | 14.632 | -121.158 | -57.269 | 137.425 |
| 31.000 | 12.919 | -116.947 | -57.482 | 139.808 |
| 32.000 | 10.779 | -111.150 | -57.931 | 142.050 |
| 33.000 | 8.100 | -102.114 | -58.619 | 144.302 |
| 34.000 | 4.862 | -85.907 | -59.561 | 146.728 |
| 35.000 | 1.980 | -55.264 | -60.789 | 149.546 |
| 36.000 | 2.166 | -16.369 | -62.354 | 153.102 |
| 37.000 | 4.534 | 8.582 | -64.341 | 158.048 |
| 38.000 | 6.733 | 21.528 | -66.878 | 165.799 |
| 39.000 | 8.356 | 28.943 | -70.074 | 179.873 |
| 40.000 | 9.494 | 33.741 | -73.340 | -151.623 |
| 41.000 | 10.255 | 37.157 | -73.632 | -109.610 |
| 42.000 | 10.718 | 39.779 | -70.982 | -81.348 |
| 43.000 | 10.933 | 41.917 | -68.410 | -67.467 |
| 44.000 | 10.936 | 43.754 | -66.484 | -59.909 |
| 45.000 | 10.749 | 45.409 | -65.087 | -55.195 |
| 46.000 | 10.383 | 46.967 | -64.086 | -51.935 |
| 47.000 | 9.846 | 48.502 | -63.389 | -49.498 |
| 48.000 | 9.137 | 50.081 | -62.935 | -47.564 |
| 49.000 | 8.247 | 51.782 | -62.683 | -45.951 |
| 50.000 | 7.163 | 53.708 | -62.606 | -44.546 |
| 51.000 | 5.857 | 56.009 | -62.684 | -43.275 |
| 52.000 | 4.288 | 58.929 | -62.905 | -42.082 |
| 53.000 | 2.393 | 62.915 | -63.260 | -40.925 |
| 54.000 | 0.076 | 68.868 | -63.745 | -39.764 |
| 55.000 | -2.780 | 78.845 | -64.359 | -38.560 |
| 56.000 | -6.095 | 97.817 | -65.104 | -37.269 |
| 57.000 | -8.381 | 132.679 | -65.989 | -35.835 |
| 58.000 | -7.156 | 169.238 | -67.025 | -34.185 |
| 59.000 | -4.529 | -170.137 | -68.229 | -32.208 |
| 60.000 | -2.285 | -159.477 | -69.630 | -29.735 |
| 61.000 | -0.573 | -153.291 | -71.267 | -26.476 |
| 62.000 | 0.726 | -149.285 | -73.196 | -21.912 |
| 63.000 | 1.717 | -146.470 | -75.485 | -15.012 |
| 64.000 | 2.476 | -144.369 | -78.172 | -3.603 |
| 65.000 | 3.054 | -142.727 | -80.970 | 16.578 |
| 66.000 | 3.487 | -141.399 | -82.454 | 48.119 |
| 67.000 | 3.803 | -140.292 | -81.333 | 78.402 |
| 68.000 | 4.022 | -139.349 | -79.173 | 96.998 |
| 69.000 | 4.159 | -138.531 | -77.201 | 107.508 |
| 70.000 | 4.228 | -137.809 | -75.615 | 113.891 |
| 71.000 | 4.238 | -137.164 | -74.362 | 118.101 |
| 72.000 | 4.199 | -136.582 | -73.369 | 121.067 |
| 73.000 | 4.118 | -136.053 | -72.577 | 123.264 |
| 74.000 | 4.001 | -135.568 | -71.944 | 124.954 |
| 75.000 | 3.855 | -135.121 | -71.437 | 126.293 |
| 76.000 | 3.684 | -134.708 | -71.035 | 127.377 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 77.000 | 3.494 | -134.326 | -70.717 | 128.270 |
| 78.000 | 3.288 | -133.973 | -70.472 | 129.015 |
| 79.000 | 3.071 | -133.648 | -70.287 | 129.642 |
| 80.000 | 2.847 | -133.349 | -70.154 | 130.173 |
| 81.000 | 2.618 | -133.076 | -70.065 | 130.623 |
| 82.000 | 2.389 | -132.829 | -70.015 | 131.004 |
| 83.000 | 2.161 | -132.609 | -69.999 | 131.326 |
| 84.000 | 1.938 | -132.416 | -70.013 | 131.594 |
| 85.000 | 1.723 | -132.251 | -70.052 | 131.814 |
| 86.000 | 1.515 | -132.115 | -70.116 | 131.990 |
| 87.000 | 1.319 | -132.008 | -70.201 | 132.123 |
| 88.000 | 1.134 | -131.931 | -70.306 | 132.218 |
| 89.000 | 0.963 | -131.885 | -70.430 | 132.274 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
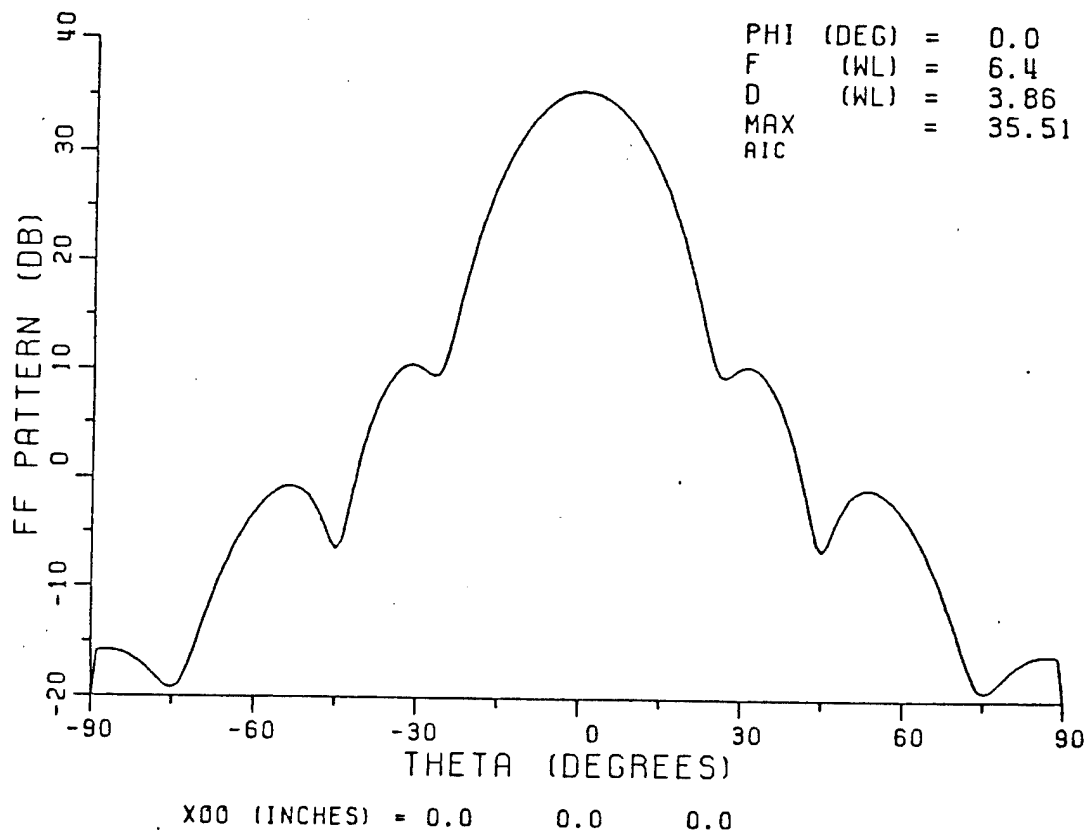
Figure 1(a). H-plane patterns of the 1.2" D conical horn.

PHI (DEG) = 45.0
F (WL) = 6.4
D (WL) = 3.86
MAX = 35.51
AIC

FF PATTERN (DB)

THETA (DEGREES)

XOO (INCHES) = 0.0    0.0    0.0
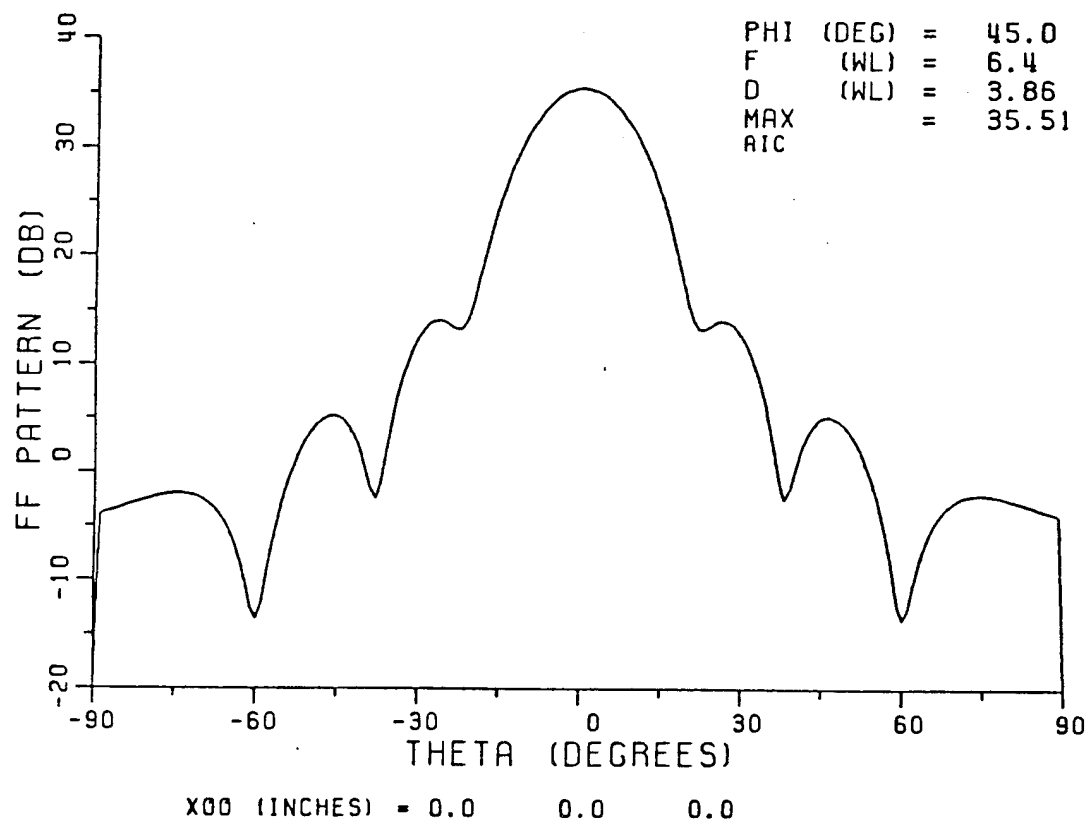
Figure 1(b).  45°-cut pattern of the 1.2" D conical horn.

11-13
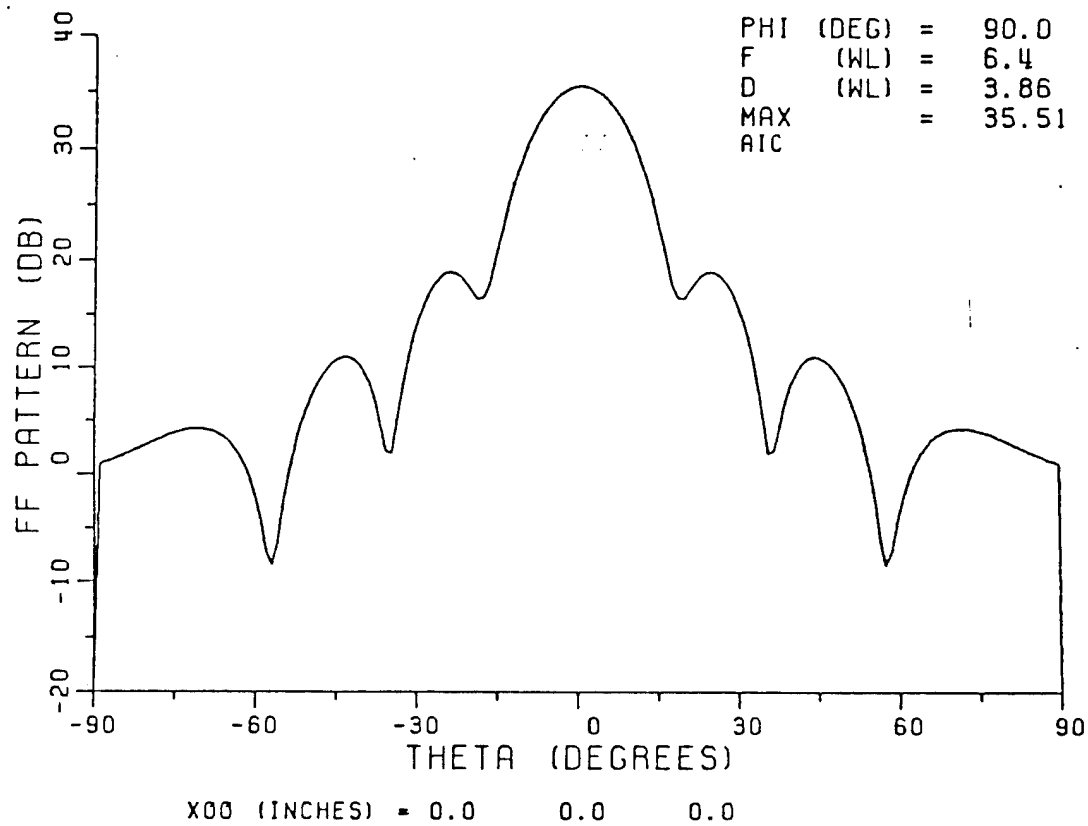
Figure 1(c).  E-plane pattern of the 1.2" D conical horn.

---
Command PA:  PLOT APERTURE DISTRIBUTION
---

     This command provides the capability of plotting aperture field
distributions which are specified by the AP: Command.  All plot data are
output on Unit #16.


BLOCK DIAGRAM FOR PLOTS OF APERTURE DISTRIBUTION


```
                        | Command PA: |
                               |
                               ↓
                   | Read:  NPA, NPAHV |
                               |
                               ↓
                | Read:  ANPA(L), L=1,NPA |
                               |
                               ↓
              | Read:  PARI, PARF, DPAR |
                               |
                               ↓
                    | Read:  NPPA |
                               |
                               ↓
    - - - - - - - →| DO NI=1,NPPA |
   |                           |
   |                           ↓
   |      | Read:  NOPA(NI), IOPA(NI) |
   |                           |
   |                           ↓
    - - - - - - - - - - - -| CONTINUE |
```
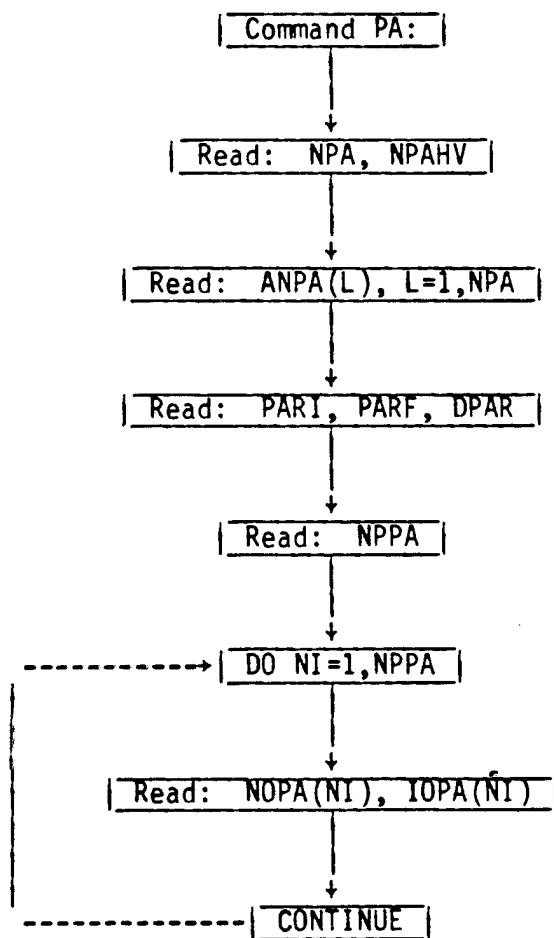
1. READ: MPA. MPAHV

    a) NPA: This integer variable specifies the number of cuts across the aperture of the reflector.

    b) NPAHV: This integer variable specifies whether the cuts are horizontal or vertical.

      NPAHV=1: horizontal cut

      NPAHV=2: vertical cut


2. READ: ANPA(L), L=1, NPA

This read statement is used to specify the locatin of the cut.

    a) ANPA(L): This real variable specifies the location of the cut. IF NPAHV=1, this variable specifies the y-coordinate (in units) of the cut; if NPAHV=2, this variable specifies the x-coordinate (in units) of the cut.


3. READ: PARI, PARF, DPAR

    a) PARI: This real variable specifies the initial x or y coordinate (in units) of the aperture points along each cut.

    b) PARF: This real variable specifies the final x or y coordinate (in units) of the aperture points along each cut.

    c) DPAR: This real variable specifies the increment (in units) in x or y coordinates of the aperture points along each cut.


4. READ: NPPA

    a) NPPA: This integer variable specifies the number of plots for each cut.

5. READ:  NOPA(NI), IOPA(NI)

This statement is executed NPPA times.

a) This integer variable specifies the polarization component to be plotted as follows:

$$\text{NOPA(NI)} = \begin{cases} 1 & : \quad \text{X component of the aperture field} \\ \\ 2 & : \quad \text{Y component of the aperture field} \end{cases}$$

b) IOPA(NZ):  This integer specifies the format to be plotted as follows:

$$\text{IOPA(NI):} \begin{cases} 1 & : \quad \text{magnitude of the aperture field} \\ 2 & : \quad \text{dB value of the aperture field} \\ 3 & : \quad \text{phase of the aperture field} \end{cases}$$

```
-----------------------------------------------------------
```
Command WG:  WAVEGUIDE MODE
```
-----------------------------------------------------------
```

This command provides an efficient calculation of the radiation patterns of some useful waveguide modes.  The analytic radiation field expressions as given in Silver, Microwave Antenna Theory and Design, page 337-338 are used to obtain the patterns.  The aperture fields of each waveguide mode have been normalized so that each mode carries unit power.

BLOCK DIAGRAM FOR WG:

```
                    ┌─────────────────┐
                    │  Command WG:    │
                    └────────┬────────┘
                             │
                    ┌────────┴────────┐
                    │  Read:  NMODE   │
                    └────────┬────────┘
                             │
                            ╱ ╲
                          ╱     ╲
                        ╱ NMODE=3 ╲
                       ╱    or      ╲      NO
                       ╲ NMODE=5?   ╱─────────────┐
                        ╲          ╱              │
                          ╲      ╱                │
                            ╲  ╱                  │
                             │                    │
                            YES                   │
                             │                    │
                    ┌────────┴────────┐           │
                    │ Read: FRTM, PHTM│           │
                    └────────┬────────┘           ↓
                             │                    │
                            ╱ ╲                   │
                          ╱     ╲                 │
                        ╱         ╲     NO         │
                       ╱ NMODE=5? ╲────────────   │
                       ╲          ╱               │
                        ╲        ╱                │
                          ╲    ╱                  │
                            ╲╱                    │
                            YES                   ↓
                             │                    │
                    ┌────────┴────────┐           │
                    │ Read: FRTE, PHTE│           │
                    └────────┬────────┘           │
                             │                    │
                             └────────────────────┘
```

13-2
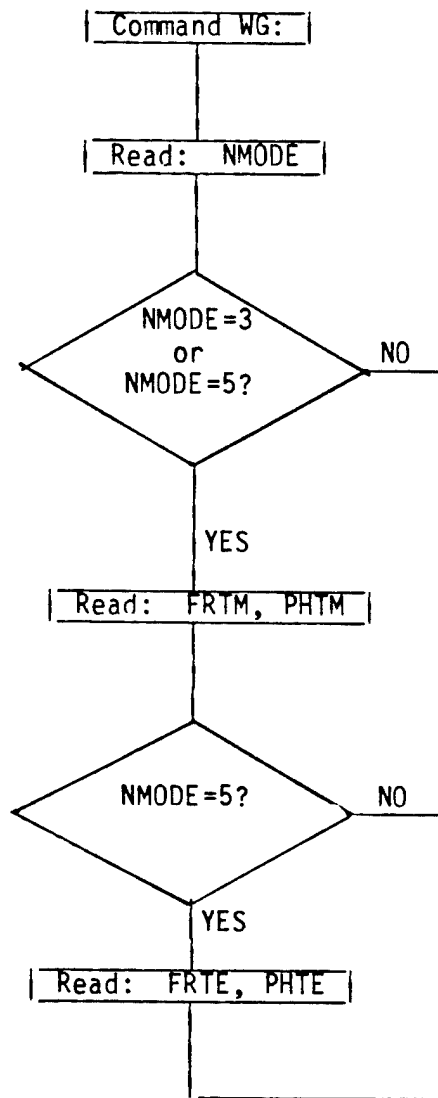
1. Read: NMODE

    a) NMODE: This integer variable specifies the type of waveguide
       mode as follows:

$$
\text{NMODE:} \begin{cases}
1 & : \text{Circular } TE_{11} \text{ mode} \\
2 & : \text{Circular } TM_{11} \text{ mode} \\
3 & : \text{Circular } TE_{11} \text{ and } TM_{11} \text{ modes} \\
4 & : \text{Circular } TE_{12} \text{ mode} \\
5 & : \text{Circular } TE_{11}, TM_{11}, \text{ and } TE_{12} \text{ modes}
\end{cases}
$$

    The field ratios between $TM_{11}$ amd $TE_{11}$ modes for NMODE=3 or
NMODE=5 is specified by the read statement 2 and field ratio
between $TE_{12}$ and $TE_{11}$ modes for NMODE=5 is given in read
statement 3.

2. Read: FRTM, PHTM

    This read statement is executed for NMODE=3 or NMODE=5 only.

    a) FRTM: This real variable specifies the magnitude of the field
       ratio ($TM_{11}/TE_{11}$).

    b) PHTM: This real variable specifies the phase shift (in degrees)
       of the $TM_{11}$ mode from the $TE_{11}$ mode.

    The complex field ratio is thus specified as

$$
\frac{E_{TM_{11}}}{E_{TE_{11}}} = FRTM \cdot e^{j \cdot PHTM}
$$

3. Read: FRTE, PHTE

This read statement is executed for NMODE=5 only.

a) FRTE: This real variable specifies the magnitude of the field ratio between the $TE_{12}$ and the $TE_{11}$ modes.

b) PHTE: This real variable specifies the phase shift (in degrees) of the $TE_{12}$ mode from the $TE_{11}$ mode.

Thus, the complex field ratio between the $TE_{12}$ and the $TE_{11}$ mode is given as

$$\frac{E_{TE_{12}}}{E_{TE_{11}}} = FRTE \cdot e^{jPHTE}$$

```
----------------------------------------------------------------
Command BR:  BODY OF REVOLUTION (MOMENT METHOD)
----------------------------------------------------------------
```

This command enables the user to calculate the radiation patterns of circular conical, corrugated and dual mode horn antennas. The geometries of these three horns are shown in Figure 1. The radiation patterns of a circular waveguide can also be calculated by this command. The geometry of the circular waveguide is shown in Figure 2. Moment Method is used to calculate the patterns. The corrugated horn patterns are simulated by the H-plane pattern of conical horn which has the same geometry as the inner dimension of the corrugated horn. The patterns of the corrugated horn are assumed to be circular symmetric.

This command is used similar to the AP: and WG: Commands. one should not use the FD: Command when this command is used. The input dimension is specified by IUNIT in the DG: Command while the other variables in the DG: Command will be ignored. If the DG: Command is not used, the input dimension is in inches.

Although one can also use either the AP: or the WG: Commands to calculate the radiation patterns of these horns, only the BR: Command can be used to generate backlobes of the patterns since aperture integration is used in the AP: and WG: Commands and only gives patterns in the forward region. Also the Aperture Integration is not accurate for small aperture horns. One can also let LFDOUT=true in the PZ: Command to output the patterns from Unit #7.

BLOCK DIAGRAM FOR BR:  COMMAND

```
            ┌─────────────────┐
            │  Command BR:    │
            └─────────────────┘
                     │
                     │
            ┌─────────────────┐
            │   Read NBR      │
            └─────────────────┘
                     │
                     │
   ┌─────────────────────────────────────────────┐
   │  Read:  FLARBR,DWG1,ALWG1,DWG2,ALWG2,TAU     │
   └─────────────────────────────────────────────┘
```
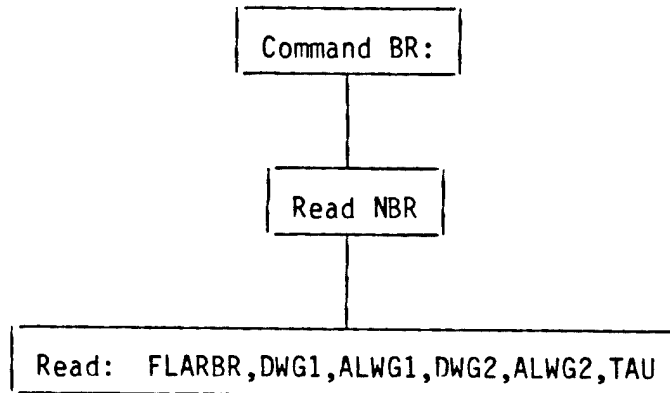
1. Read:  NBR

   a) NBR:  This integer variable specifies the type of the horn as
      follows:

      NBR=3   :   Conical Horn

      NBR=4   :   Corrugated Horn

      NBR=5   :   Dual Mode Horn

      NBR=6   :   Circular Waveguide


2. Read:  FLARBR, DWG1, ALWG1, DWG2, ALWG2, TAU

   a) FLARBR:  This real variable specifies the full flare angle (in
      degrees) of the flared section of the horn.  Note FLARBR should
      not be zero.  When NBR=6, this variable is ignored.

   b) DWG1:  This real variable defines the aperture diameter of the
      horn.  When NBR=6, this variable is ignored.

   c) ALWG1:  This real variable specifies the length of the waveguide
      section of the dual-mode horn where the diameter of the
      waveguide is DWG1.  For conical or corrugated horns, this
      variable is ignored in the code.  When NBR=6, this variable is
      also ignored.

   d) DWG2:  This real variable specifies the diameter of the
      waveguide which is used to generate the circular waveguide $TE_{11}$
      mode.  In order to assure that the only waveguide mode
      propagating is the $TE_{11}$ mode, DWG2 must be small enough so that
      higher order modes are cut off.  On the other hand, DWG2 must be
      large enough to let the $TE_{11}$ mode propagate.  Normally,
      $0.586\lambda <$ DWG2 $< 1.220\lambda$.  A typical number for DWG2 is $0.75\lambda$.
      For NBR=6, this variable specifies the diameter of the
      waveguide.

   e) ALWG2:  This real variable specifies the waveguide length where
      the waveguide is used to generate the $TE_{11}$ mode and has diameter
      DWG2.

   f) TAU:  This real variable defines the polarization angle of the
      horn.

NBR = 3

CONICAL HORN

NBR = 4

CORRUGATED HORN

NBR = 5
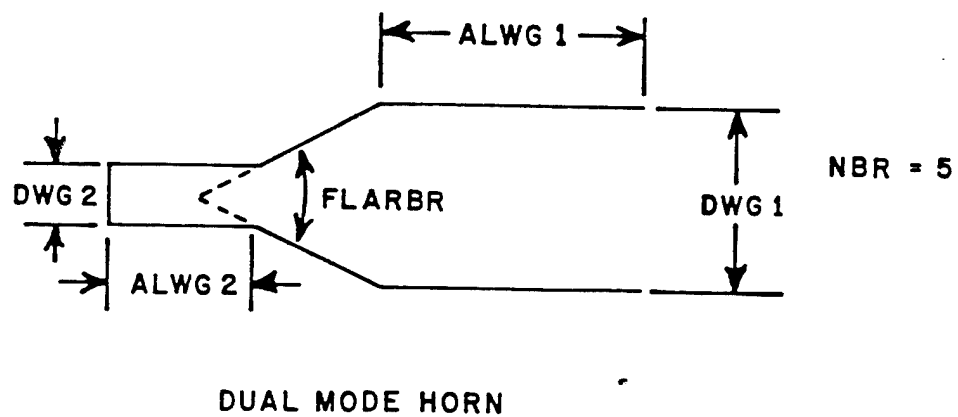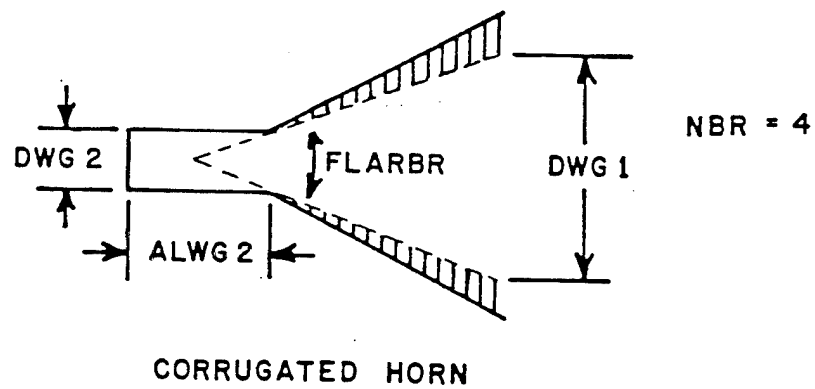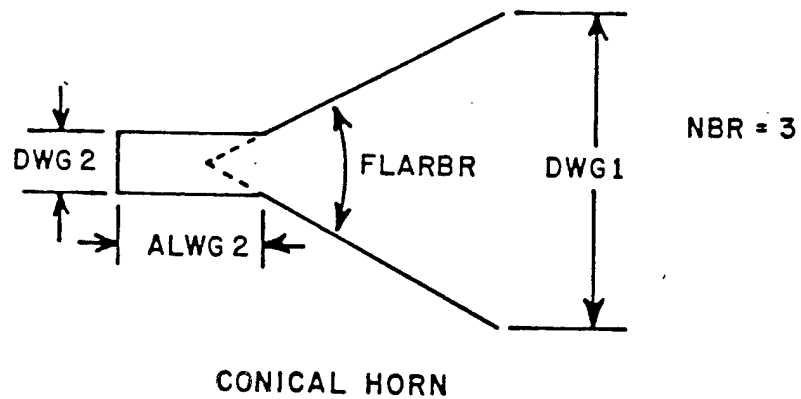
DUAL MODE HORN

Figure 1.  Geometries of 3 circular symmetric horns.
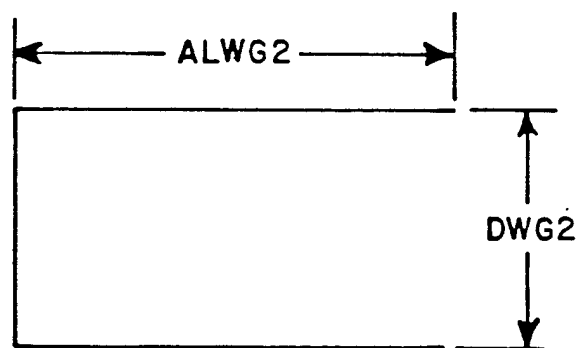
Figure 2.  Geometry of a circular waveguide.

Example 1:

In this example, the radiation patterns of a dual-mode horn are calculated by the BR: Command. The frequency is 11.0 GHz. The calculated patterns are given in Figure 3 and the measured patterns are given in Figure 4. Note that the backlobe in the calculation is higher than the measurement because structures for mounting and feeding the horn in the measurement system are not modeled in the calculation. The input data for the calculation are given in Table 1.
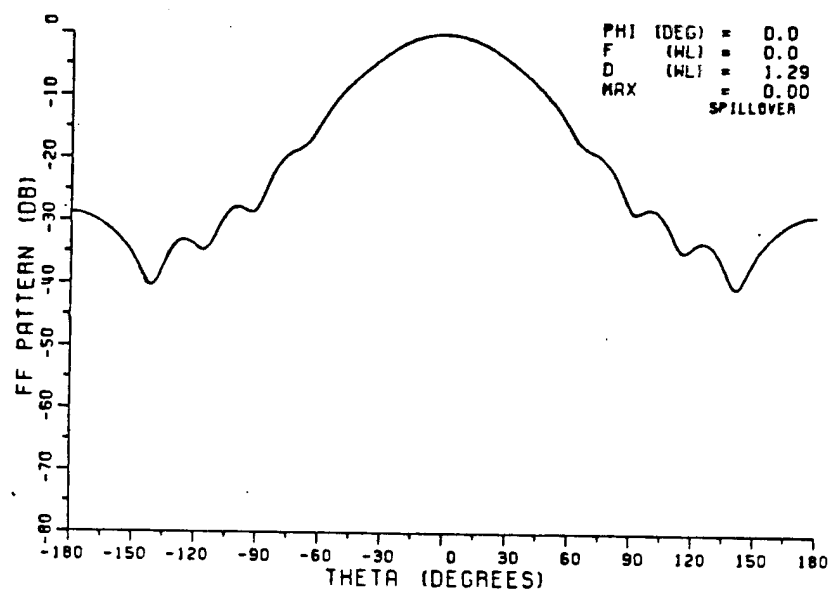
(a) H-plane
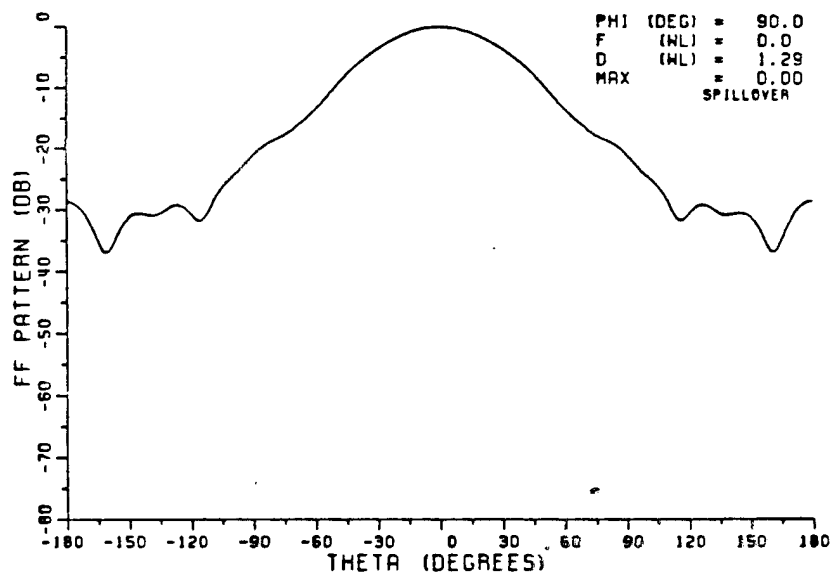


(b) E-plane

Figure 3. Calculated patterns of the dual mode horn.

14-7
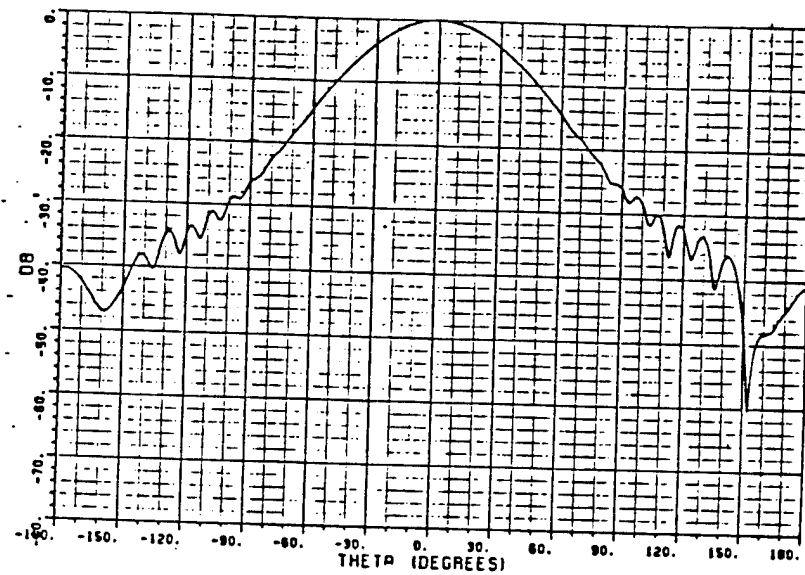
(a) H-plane
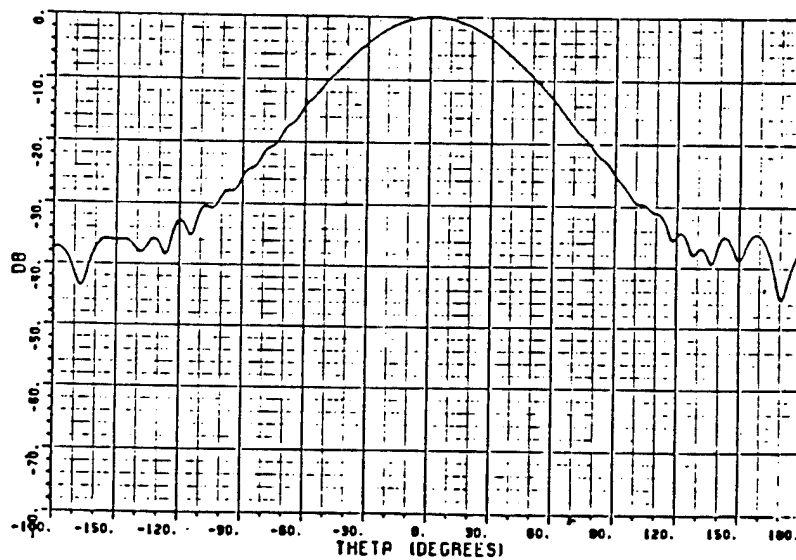


(b) E-plane

Figure 4.  Measured patterns of the dual mode horn.

# TABLE 1

## INPUT DATA FOR PATTERN CALCULATION OF A DUAL MODE HORN

```
CM:          ***** BR5.DAT *****
CE: MOMENT METHOD FOR DUAL MODE HORN
FQ:
1 11.
BR:   NBR = 5 : DUAL MODE HORN
5
60. 1.384 1.447 0.75 1. 90.
PZ:   LFDOUT = F
3
0.   45.   90.
-180. 180. 1.
F
PP:
1
1  1
1  2
XQ:
```

Example 2:

In this example, the radiation patterns of a conical horn are calculated at 11.0 GHz. The calculated and measured patterns are given in Figures 5 and 6, respectively. The input data are given in Table 2.
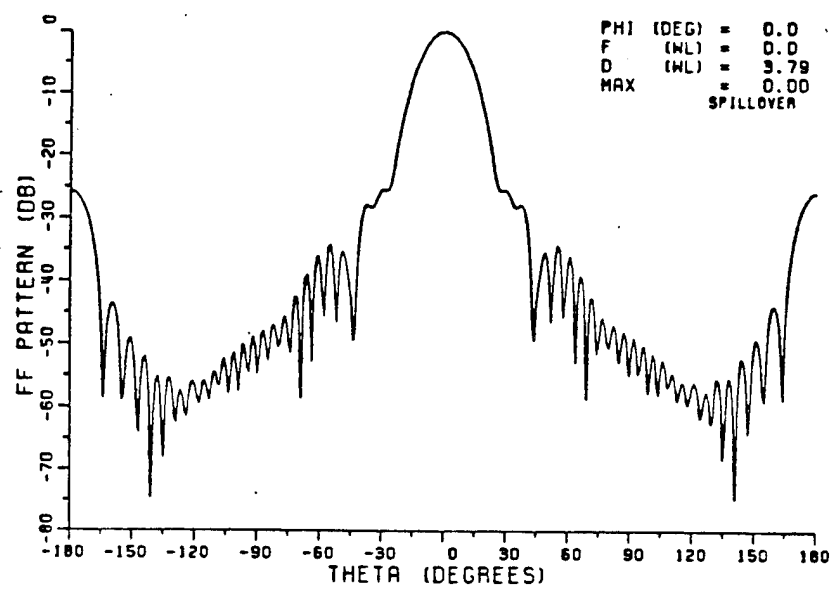
(a) H-plane
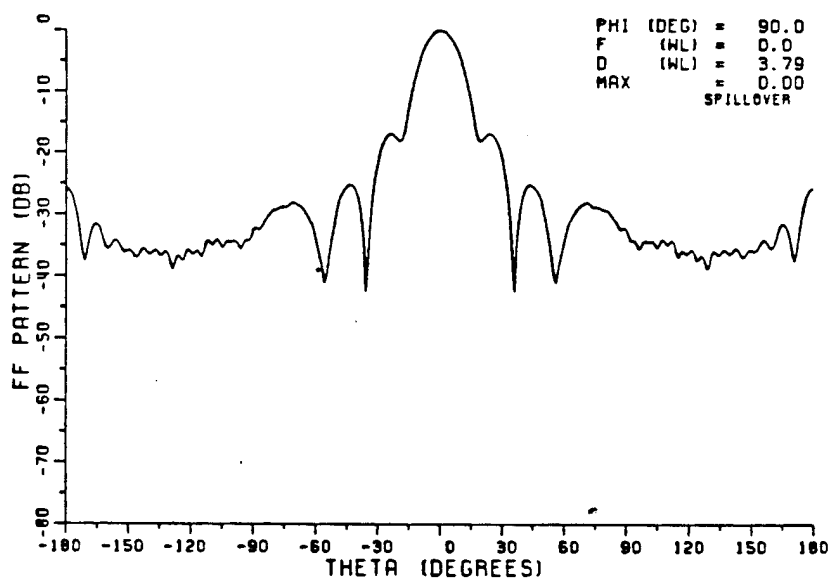


(b) E-plane

Figure 5.  Calculated patterns of a conical horn.
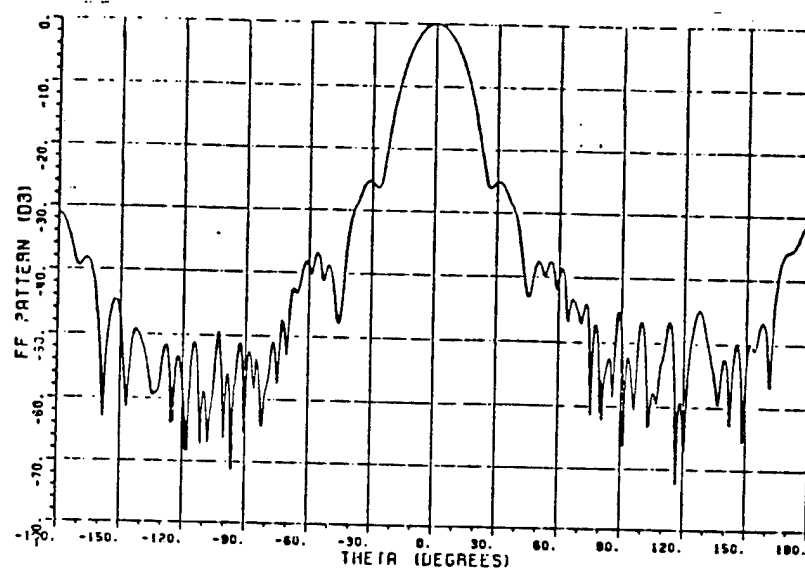
(a) H-plane
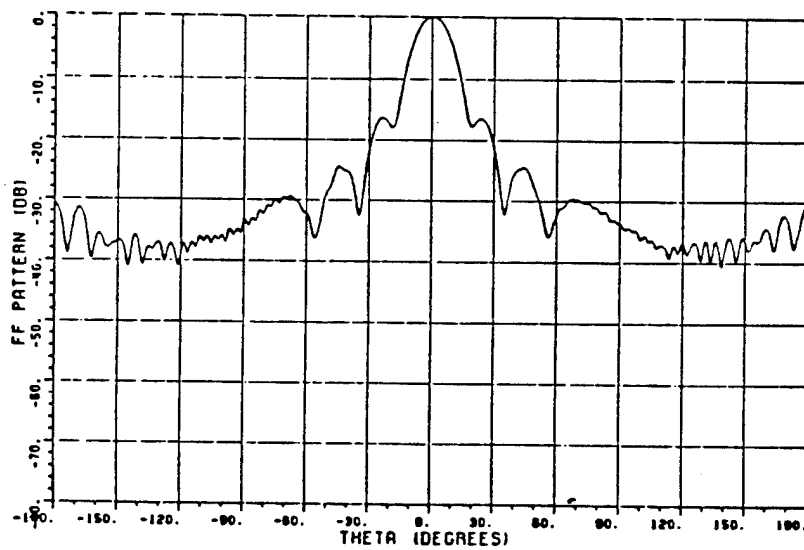


(b) E-plane

Figure 6. Measured patterns of the conical horn.

TABLE 2

INPUT DATA FOR PATTERN CALCULATION FOR A CONICAL HORN

```
CM:  ***** BR3_CASS.DAT *****
CE: MOMENT METHOD FOR CONICAL HORN
FQ:
1 11.
BR:  NBR = 3 : CONICAL HORN
3
15. 4.072  0. 0.75 1. 90.
PZ:  LFDOUT = F
3
0.  45.  90.
-180. 180. 1.
F
PP:
1
1  1
1  2
XQ:
```

```
----------------------------------------------------------
Command FQ:  FREQUENCY
----------------------------------------------------------
```

This command enables the user to specify the frequencies for which patterns are to be computed.

```
              _____
             |  Command FQ:      |
             |_____|
                      |
                      |
    _____
   |  Read:   NFRQ,(FREQ(I),I=1,NFRQ)         |
   |_____|
```
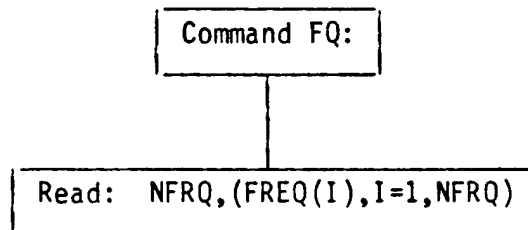
1.  READ:  NFRQ,(FREQ(I),I=1,NFRQ)

   a) NFRQ:  This integer variable is used to define the number of
      frequency inputs.  If the feed is frequency dependent, use only
      one input frequency FREQ(1) (NFRQ=1) in conjunction with a new
      input feed pattern for each frequency.  Thus a new FD:  Command
      must be used for each frequency with a frequency-dependent feed.
      Presently 1 < NFRQ < 10.

   b) FREQ(I):  This is a dimensioned real variable which defines the
      Ith frequency (in GHz) for which a given antenna design with a
      frequency independent feed pattern is to be run.

----------------------------------------------------------
Command FD:  FEED PATTERN
----------------------------------------------------------

   This command enables the user to specify the feed pattern.  Feed
patterns can be specified by piecewise linear feed data, analytic
functions, or by feed horn dimensions.

NOTE:  For offset reflectors, the TL:  Command must also be used to tilt
the feed axis.

# BLOCK DIAGRAM FOR FEED PATTERN

```
                        ┌─────────────────┐
                        │  Command FD:    │
                        └─────────────────┘
                                 │
                                 V
                        ┌─────────────────────┐
                        │ Read:  NHORN,LFOCUS  │
                        └─────────────────────┘
                                 │
                               ╱   ╲          YES
                              ╱     ╲    ------->-----
                             ⟨ NHORN>1? ⟩              │
                              ╲     ╱                  │
                               ╲   ╱                   │
                                 │ NO                  │
                                 │          ┌──────────────┐
                                 │          │ Read:  LSI   │
                                 │          └──────────────┘
                                 V <-------------------
                               ╱   ╲          TRUE
                              ╱     ╲    ----->--------
                             ⟨IF (LFOCUS)⟩              │
                              ╲     ╱                   │
                               ╲   ╱                    │
                                 │ FALSE                │
                                 V                      │
                        ┌─────────────────────┐        │
                        │ Read:  (DXS(N),N=1,3)│        │
                        └─────────────────────┘        │
                                 │ ---------------------│
                               ╱   ╲          NO
                              ╱     ╲    ------->-----
                             ⟨ NHORN=0? ⟩              │
                              ╲     ╱                  │
                               ╲   ╱     ┌───────────────────────────────────────┐
                                 │ YES   │READ: ANG1,AW1,ANG2,AW2,AD,TAU,LCALAD  │
                                 │       │         For horn feed                 │
                                 │       └───────────────────────────────────────┘
                                 V
                        ┌─────────────────────────────────────┐
                        │ Read:  LLFD,NCP,LDB,ISYM,TAU,AD,LCALAD│
                        └─────────────────────────────────────┘
                        ┌─────────────────────────────────────┐
                        │ Read:  NPHI,(PHIN(N),N=1,NPHI)      │
                        └─────────────────────────────────────┘
                                 V
              TRUE             ╱   ╲           FALSE
        -------------------   ⟨If(LLFD)⟩   --------------
                    │          ╲   ╱                     │
                    │           ╲ ╱                      │
          ┌──────────────┐                     ┌──────────────────────────────────────────┐
          │ Read:  N2    │                     │Read: NPW,(AEX(N),CAN(N),PSIO(N),N=1,NPHI)│
          └──────────────┘                     │          For analytic pattern            │
                    │                          └──────────────────────────────────────────┘
    ┌──────────────────────────┐           ┌──────────────────────────────────┐
    │Read: PSIX,FN,PHS,FNX,PHSX│           │ Read:  PSL1(N),PSL2(N),N=1,NPHI │
    │For piece-wise linear     │           └──────────────────────────────────┘
    │pattern                   │
    └──────────────────────────┘
```

16-2

1. READ: NHORN,LFOCUS

    a) NHORN: This integer variable specifies whether a horn feed is to
       be used or not. If a horn feed is to be used it specifies
       whether it is a regular horn or a corrugated horn as follows:

        NHORN =0        SAMPLED OR ANALYTICAL FEED INPUT
        NHORN =1        REGULAR HORN FEED
        NHORN =2        CORRUGATED HORN FEED

    b) LFOCUS: This logical variable is used to tell the code whether
       the feed is located at the focus or not.

        LFOCUS =TRUE    FEED IS FOCUSED
        LFOCUS =FALSE   FEED IS DEFOCUSED


2. READ: LSI

    This statement is executed for NHORN>1 only.

    a) LSI: This logical variable gives the option to use a line source
       integration (AI) in the forward half of the H-plane pattern for
       ordinary horn feeds in the FD: Command (NHORN>1). If LSI is
       false, as in the default data, GTD is used for all angles in the
       H-plane pattern (normally set false).


3. READ: (DXS(N),N=1,3)

    This statement is skipped if LFOCUS=true.

    a) DXS(N): This dimensioned real variable is used to specify the
       displacement of the feed from the focus.

4. READ: LLFD,NCP,LDB,ISYM,TAU,AD,LCALAD

This statement is skipped if NHORN is not 0.

a) LLFD: This logical variable is used to tell the code whether or
not a piecewise linear feed pattern is to be used. If set false
an analytic function is used.

| | |
|---|---|
| LLFD=TRUE | INPUT FEED PATTERN IN TERMS OF LINEAR DATA POINTS |
| LLFD=FALSE | ANALYTIC FUNCTION |

b) NCP: This integer variable is used to tell the code whether or
not the feed is circularly polarized.

| | |
|---|---|
| NCP=0 | FEED IS LINEARLY POLARIZED |
| NCP=-1 | FEED IS LEFT CIRCULARLY POLARIZED |
| NCP=+1 | FEED IS RIGHT CIRCULARLY POLARIZED |

c) LDB: This logical variable is used to tell the code whether or
not the feed pattern input and output data are specified in dB or
not. If LDB is false, feed pattern input and output are linear
field values.

| | |
|---|---|
| LDB=TRUE | FEED DATA INPUT IN DB |
| LDB=FALSE | LINEAR FEED DATA INPUT |

d) ISYM: This integer variable defines the type of symmetry for the
feed pattern. Positive values are used for even symmetry (sum
patterns) and negative values are used for odd symmetry
(difference patterns). The absolute value (IB= $|$ISYM$|$) defines
the regions of symmetry with respect to the feed coordinate
system (x,y,z) shown in Figure 1.

| | |
|---|---|
| $|$ISYM$|$=IB=0: | No symmetry |
| $|$ISYM$|$=IB=1: | Symmetry with respect to x and y axes |
| $|$ISYM$|$=IB=2: | Symmetry with respect to x axis |
| $|$ISYM$|$=IB=3: | Symmetry with respect to y axis |

e) TAU: This real variable is input in degrees and defines the
linear polarization angle relative to the x-axis of the feed as
shown in Figure 1.

| | |
|---|---|
| TAU=0 | FOR HORIZONTAL POLARIZATION |
| TAU=90 | FOR VERTICAL POLARIZATION. |

16-4

f) AD: This variable defines the distance of the horn aperture from the feed reference point XS (see statement 9).

g) LCALAD: This logical variable specifies that whether the phase center of the feed is to be calculated by the code or not. If AD≠0, this variable is ignored in the code (but still has to be input). In the Cassegrain reflector pattern calculation, LCALAD should be false when the subreflector patterns are used as feed patterns to calculate the main reflector patterns.
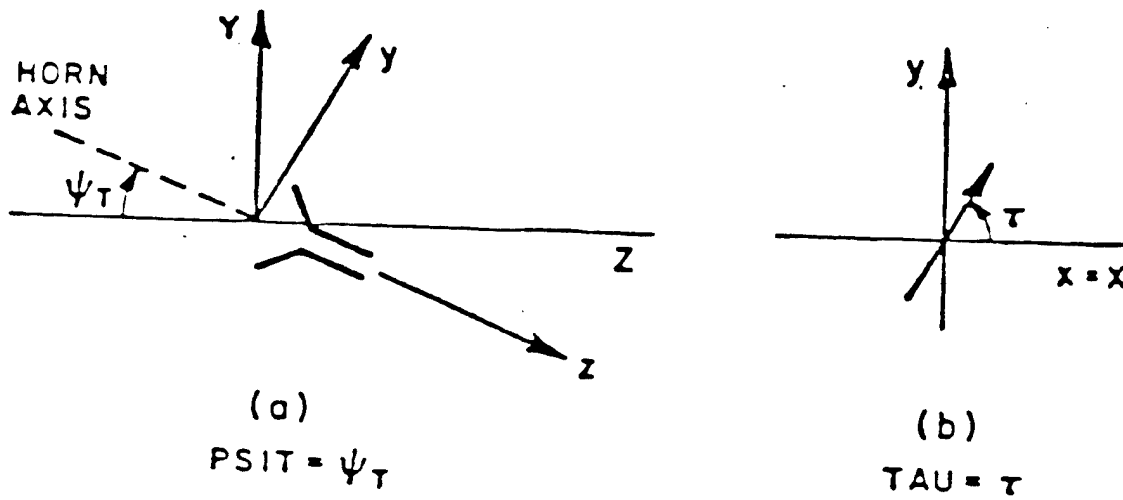
Figure 1. Coordinate system of feed horn and polarization angle tau when linearly polarized.
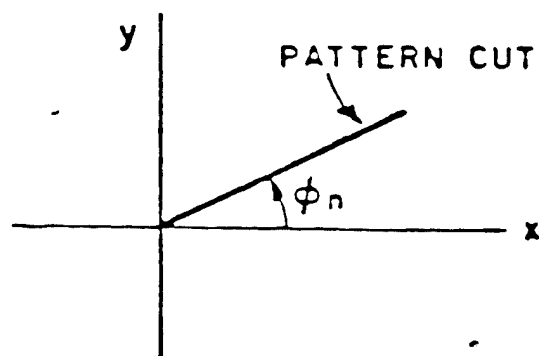


Figure 2. N-th input feed pattern cut, PHIN(N)=$\phi_n$.

4. READ: NPHI, (PHIN(N),N=1,NPHI)

This statement is skipped if NHORN is not 0.

a) NPHI: This integer variable defines the number of input feed pattern cuts. Each input pattern corresponds to a $\phi$-plane cut with respect to the feed axis (same as reflector axes for PSIT=0.) A circularly symmetric feed pattern is obtained if NPHI=1 and ISYM=1. Presently, 1<NPHI<15.

b) PHIN(N): This dimensioned real variable is input in degrees and defines the $\phi$n angle of the N-th pattern cut as shown in Figure 2. The values must be input in monotonic order, i.e., PHIN(N+1)>PHIN(N). The first value N=1 and the last value N=NPHI must be consistent with the type of pattern symmetry as shown below:

| IB=\|ISYM\| | Type of Symmetry | PHIN(1) | PHIN(NPHI) |
|---|---|---|---|
| 0 | None | -180° | <180° |
| 1 | x&y axes | 0° | 90° |
| 2 | x-axis | 0° | 180° |
| 3 | y-axis | - 90° | 90° |

If ISYM=-2 or -3, the first and last pattern cuts, i.e., PHIN(1) and PHIN(NPHI), should be null patterns. Consequently, at least 3 pattern cuts (NPHI>2) are needed for ISYM=-2 or -3. For a feed pattern with no $\phi$-symmetry (ISYM=0) the -180 degree pattern cut is also automatically stored as the +180 degree pattern cut. Presently 1<N<15.

16-7
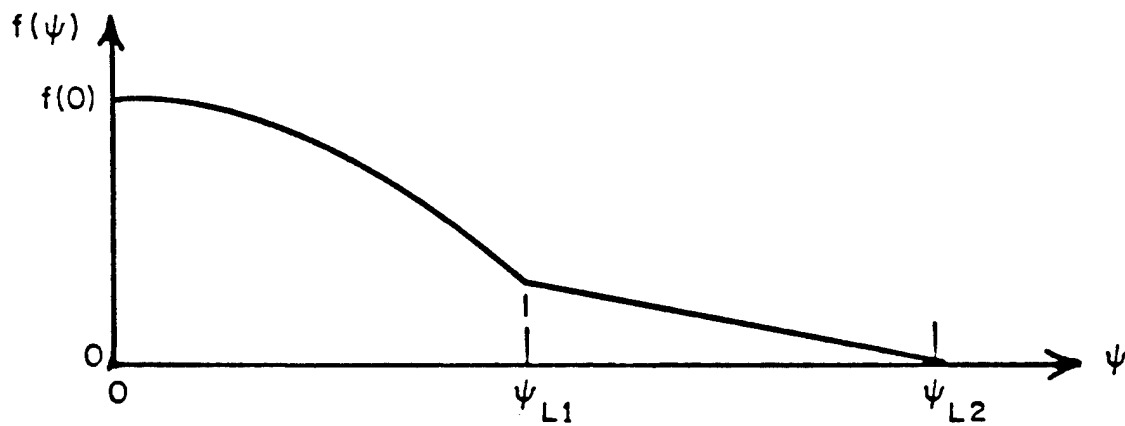
Figure 3.  Analytic feed pattern with linear taper region.

## *** ANALYTIC FEED PATTERN ***

5. READ: NPW,(AEX(N),CAN(N),PSIO(N),N=1,NPHI)

This statement is used to specify the analytic pattern (LLFD=false). The analytic functions are described in Appendix A. This read statement is skipped if LLFD=true, or if NHORN is not 0.

a) NPW: This integer variable defines the power for the cosine or sine function.

b) AEX(N),CAN(N),PSIO(N): These are dimensioned real variables which define the analytic pattern in the N-th $\phi$-pattern cut. Presently 1<N<15.


6. READ: LPSIL,(PSL1(N),PSL2(N),N=1,NPHI)

This statement is used to specify the region of linear tapering for analytic feed (LLFD=false). This statement is skipped if LLFD=true, or if NHORN is not 0.

a) LPSIL: This logical variable specifies whether linear tapering is used or not. (The default of LPSIL is false.)

b) PSL1(N): This dimensioned real variable specifies the starting angle of the linear taper in the $N^{th}$ $\phi$-pattern cut as shown in Figure 3.

c) PSL2(N): This dimensioned real variable specifies the angle at which the linear taper ends in the $N^{th}$ $\phi$-pattern cut as shown in Figure 3. Past this angle, the field is zero.


## *** LINEAR FEED INPUT PATTERN ***

7. READ: N2

This read statement is skipped if LLFD=false, or if NHORN is not 0.

a) N2: This integer variable defines the number of feed pattern points to be read for all input $\phi$-plane pattern cuts. It is used only for piecewise linear feed pattern input (LLFD=true). Presently N2<377.

8. READ: PSIX,FN,PHS,FNX,PHSX

This read statement is skipped if LLFD=false, or if NHORN is not 0. This read statement is executed N2 times for each value of PHIN(N), i.e., each PHI-cut.

a) PSIX: This real variable defines the K-th angle (in degrees) for the piecewise linear feed input pattern as shown in Figure 4. Presently up to 377 points can be specified for each PHIN-cut.

NOTE: PSIX VALUES SHOULD BE IN MONOTONIC ORDER AND THE INITIAL PSIX VALUE FOR EACH PHIN-CUT SHOULD BE PSIX=0. 0<PSIX≤180.

b) FN: This real variable defines the feed pattern value (in dB for LDB=true, or linear field value for LDB=false). Negative values may be used.

c) PHS: This real variable is the phase of the feed input pattern in degrees.

d) FNX: This real variable defines the cross-polarization feed pattern value (in dB for LDB=true, or linear field for LDFB=false). Negative values may be used. If the cross-polarization field is not available, FNX=a large negative number (e.g., -300 dB) if LDB=true or FNX=0 if LDB=false.

e) PHSX: This real variable is the phase in degrees of the cross-polarization field of feed.
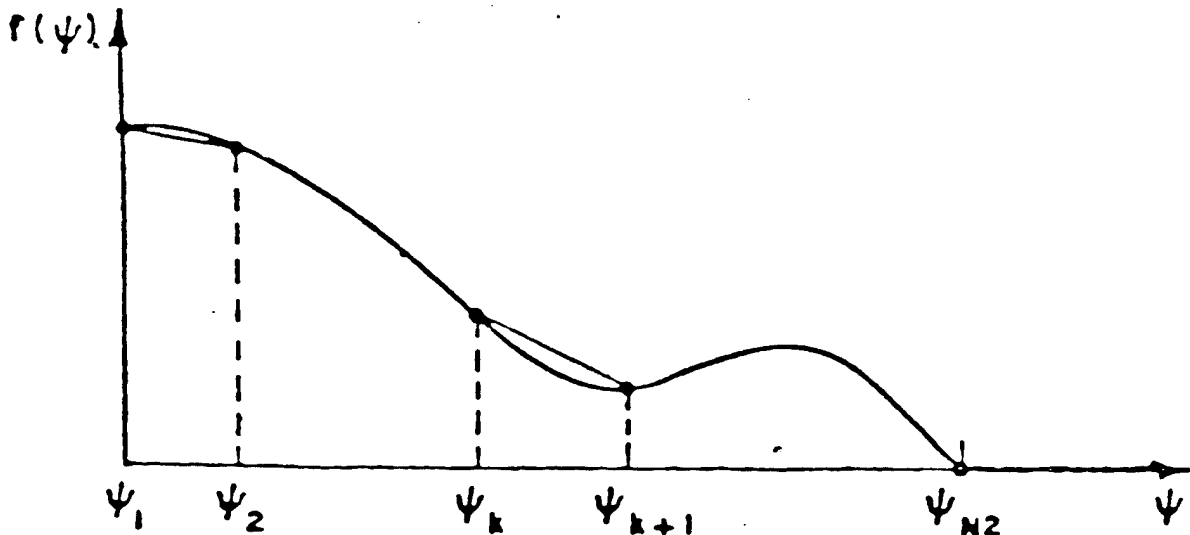


Figure 4.  Piecewise linear approximation for feed patterns.

### *** HORN FEED GEOMETRY INPUT ***

9. READ:  ANG1,AW1,ANG2,AW2,AD,TAU,LCALAD

   This statement is skipped if NHORN=0

   a) ANG1,ANG2:  These real variables are input in degrees and define
      the full horn flare angles in the PHI=0,90 planes, respectively,
      as shown in Figure 5a.

   b) AW1,AW2:  These real variables are input in the units specified
      by the variable IUNIT in the DG:  Command.  They define the horn
      aperture widths in the PHI=0,90 planes, respectively, as shown in
      Figure 5a.

   c) AD:  This real variable is input in the units specified by the
      variable IUNIT in the DG:  Command.  It defines the distance of
      the horn aperture from the feed reference point, XS, as measured
      along the horn axis.  See Figure 5a.

   d) TAU:  This real variable is input in degrees and defines the
      linear polarization angle relative to the x-axis of the feed as
      shown in Figure 1b.  TAU=0 for horizontal polarization, and
      TAU=90 for vertical polarization.
      NOTE:  TAU MUST BE EITHER 0. OR 90. FOR HORN FEEDS

   e) LCALAD:  See READ:  Statement 4.
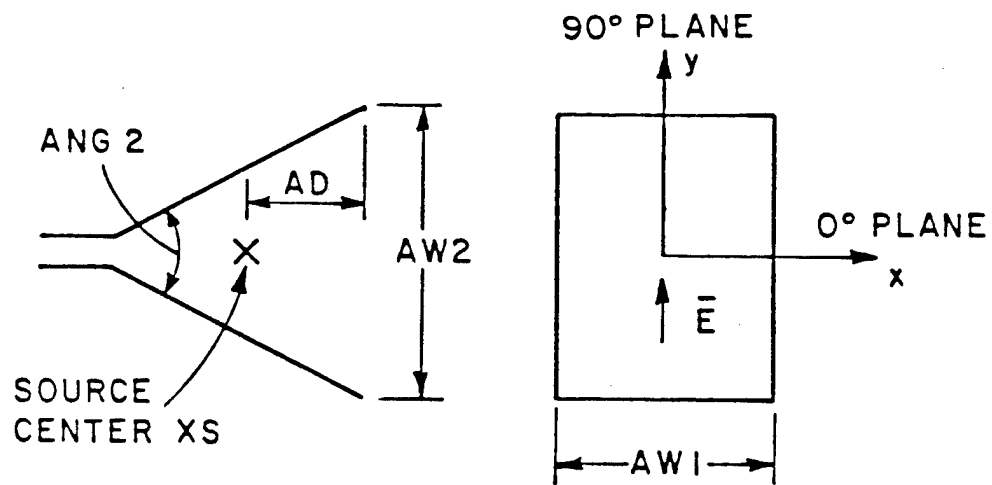
a) Rectangular horn (Regular or Corrugated)
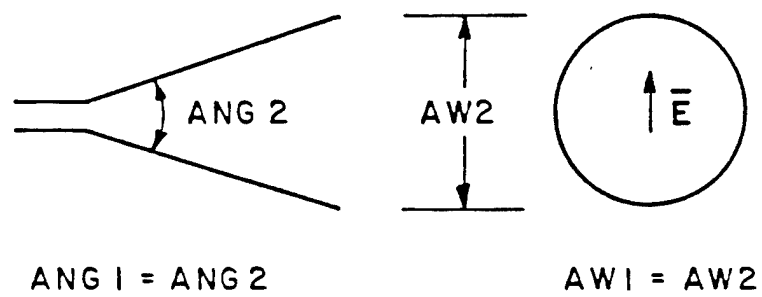


b) Conical horn (Corrugated only)

Figure 5.  Horn feed geometries.

```
-----------------------------------------------------------
Command TL:  OFFSET REFLECTOR GEOMETRY
-----------------------------------------------------------
```

This command enables the user to specify the tilt angle of the feed and the aperture center of the reflector on the Y-axis. This information is primarily useful for off-set reflectors.

BLOCK DIAGRAM FOR FEED TILT

```
            | Command TL: |
                   |
                   |
        | Read:    PSIT,YC |
```
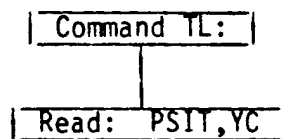
PSIT  = Feed Axis Tilt
YC    = OFFSET for Circular Reflectors

1.  READ:  PSIT,YC

   a) PSIT:  This real variable is input in degrees and defines the
      angle by which the feed horn is tilted, in the y-z plane, from
      the negative z-axis, as shown in Figure 1.

   b) YC:  This real variable is input in the units specified by the
      variable IUNIT and defines the aperture center of an offset
      reflector antenna.  It is used only for circular rim shapes in
      which case the rim points are calculated from the reflector
      diameter D.  Set YC=0 for non-circular rim shapes.
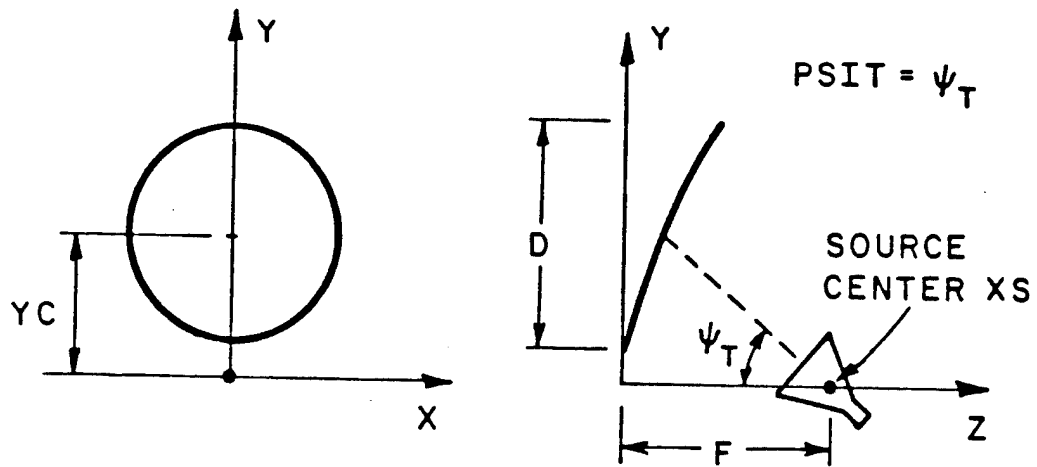
OFFSET REFLECTOR GEOMETRY (DG:,TL:)
(CIRCULAR: D > 0)

(a) Circular: D>0
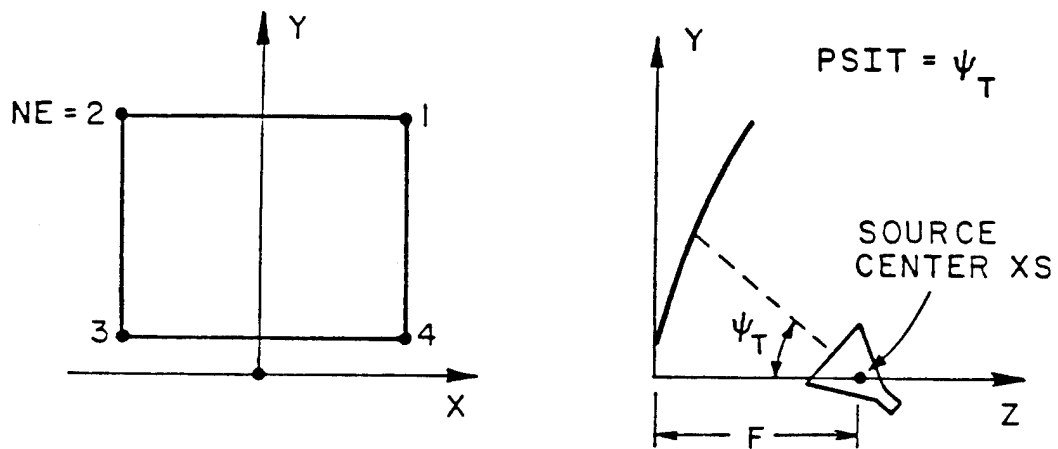


RIM (NE,1) = $X_{NE}$

RIM (NE,2) = $Y_{NE}$

OFFSET REFLECTOR GEOMETRY (DG:,TL:)
(NON-CIRCULAR: D=0)

(b) Non-circular: D=0

Figure 1. Offset reflector geometry (DG:,TL:)

```
-----------------------------------------
COMMAND AF:  ARRAY FEED
-----------------------------------------
```

This command enables the user to input the information associated with the array feed.

1.  Read: NFEED

    a) NFEED:  This integer variable is used to define the number of elements for the array feed.  Currently, 1<NFEED<101.

2.  Read: (XARO(N),N=1,3)

    b) XARO(N):  This dimensional real variable is used to specify the origin of the feed coordinate system in the reflector coordinate system.

The following two read statements are executed NFEED times.

3.  Read: (DXSAF(N,NF),N=1,3)

    a) DXSAF(N,NF):  This dimensional real variable is used to specify the location of the array feed element in the feed coordinate system.  In other words, it specifies the displacement from the origin XARO(N).  An example of a 4-element array is shown in Figure 1.

4.  Read: CRA(NF)

    a) CRA(NF):  This dimensional complex variable is used to specify the complex weights of the array elements. Currently, all the array elements have the same pattern with different weights.  The pattern is input in the FD:  command and the weights are specified by this complex variable CRA(NF).

5.  Read:  PHRE, PSTL, PHFD

    a) PHRE:  This real variable defines the phi plane in the reflector coordinate system in which the array feed is tilted.

b)  PSTL:  This real variable is used to define the angle in degrees that the array feed is tilted.

c)  PHFD:  This real variable defines the phi plane in the feed coordinate system in which the array feed is tilted.

Note that all the array elements have the same orientation defined by PHRE, PSTL, and PHFD, although it is not required to have all the elements on the same plane.  Figure 2 shows the orientation of the array feed for a general case where PHRE, PSTL, and PHFD are arbitrarily specified.  Figure 3 shows a common case where PHRE=90° and PHFD=90°.

There are two methods to get the reflector pattern if the array feed is used.  The first method (method A) uses the AF: command to calculate the reflector pattern directly, and the second method (method B) uses the AF: command to calculate the far field pattern of the array feed as the first step of a two-step procedure.

Currently, method A is good only in the AI region, and it is more time-consuming because the ray tracing technique is applied to each element of the array feed to calculate the aperture field contribution.

Method B with the two-step procedure is recommended if the reflector is located in the far field region of the array feed.  In the first run, the AF: command is used to specify the array and element locations and excitations.  Also, the logical input LFDOUT is set true in the PZ: command so that the array feed pattern is stored during the first run.  The PF: command for plotting feed patterns is borrowed to specify the number and angles of the phi-cut for the array feed pattern. After the first run, the array feed pattern is stored in the write unit #7, which should be edited to provide a suitable FD: command for the input data of the second run.  In summary, the two-step procedure (when the reflector is in the far field of array feed) is given by:

1.  The feed pattern of the array is calculated and stored by using LFDOUT=true.

2.  The reflector pattern is calculated by using the array feed as a single composite feed.

If the aperture distribution of each array element is a known function, such as $TE_{11}$ circular mode, the AP: command can be used to generate the element pattern for the array feed.  This can be done by setting LFDOUT to be true and using the AP: command to run the reflector code prior to either method A or method B.

NOTE:  When LGTD = true or strut, plate and feed blockage are included in the calculation, method B must be used.

18-2
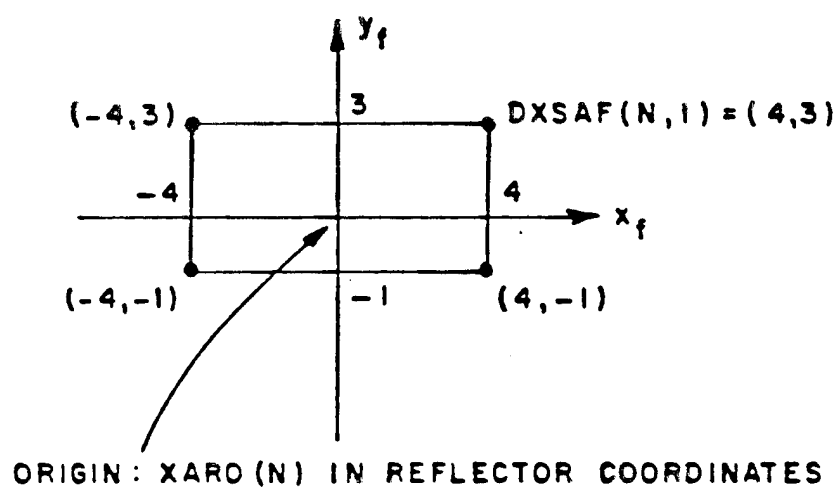
Figure 1.  An example of 4-element array feed.
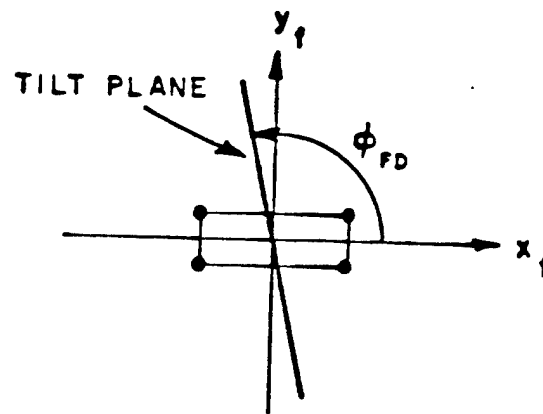
(a) $\phi_{FD}$ SPECIFIES THE TILT PLANE FOR THE ARRAY FEED.
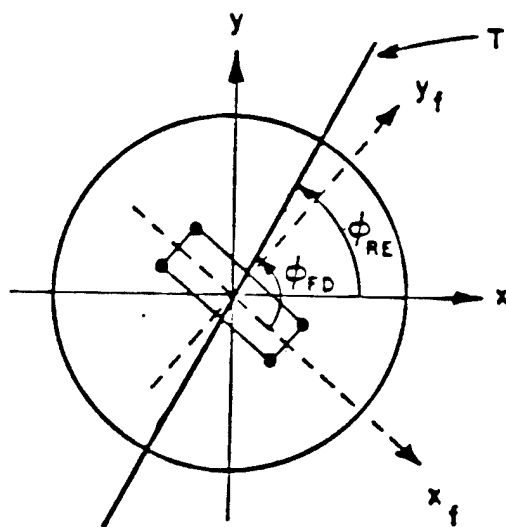
(b) FRONT VIEW.
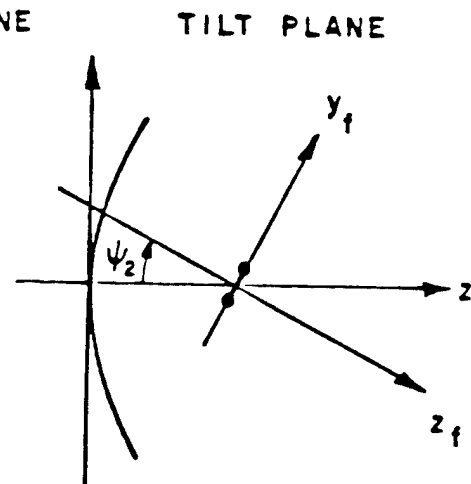
(c) SIDE VIEW FOR TILT PLANE ($\phi = \phi_{RE}$).

Figure 2. The orientation of the array feed for a general case. ($\psi_2$ = PSTL)
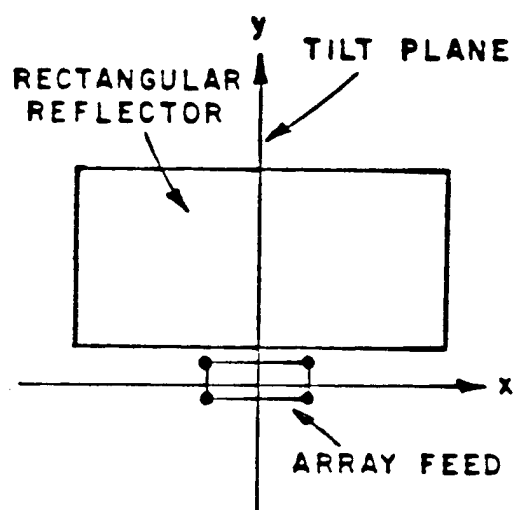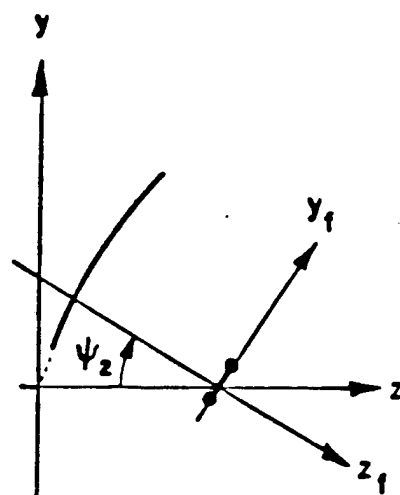
Figure 3. A common case where $\phi_{RE} = 90°$ and $\phi_{FD} = 90°$.

```
                        | Command AF:|
                             |
                             |
              | Set array feed logic true  |
              |          LAF=.TRUE.         |
                             |
                             |
              |Read number of array elements|
              |            NFEED            |
                             |
                             |
                 | Read: (XARO(N),N=1,3)|
                             |
                             |
              | Loop through array elements |<-----
              |        NF=1,  NFEED         |      |
                             |                     |
                             |                     |
               | Read: (DXSAF(N,NF),N=1,3)|        |
                             |                     |
                             |                     |
                   | Read: CRA(NF)|                |
                             |                     |
                       ------>---------------------
                             |
                  | Read: PHRE, PSTL, PHFD |
                             |
                             |
              |   Call subroutine ROTL to   |
              | calculate unit vectors VROT |
                             |
                             |
              |    Transform feed locations   |
              | to reflector coordinate system|
```
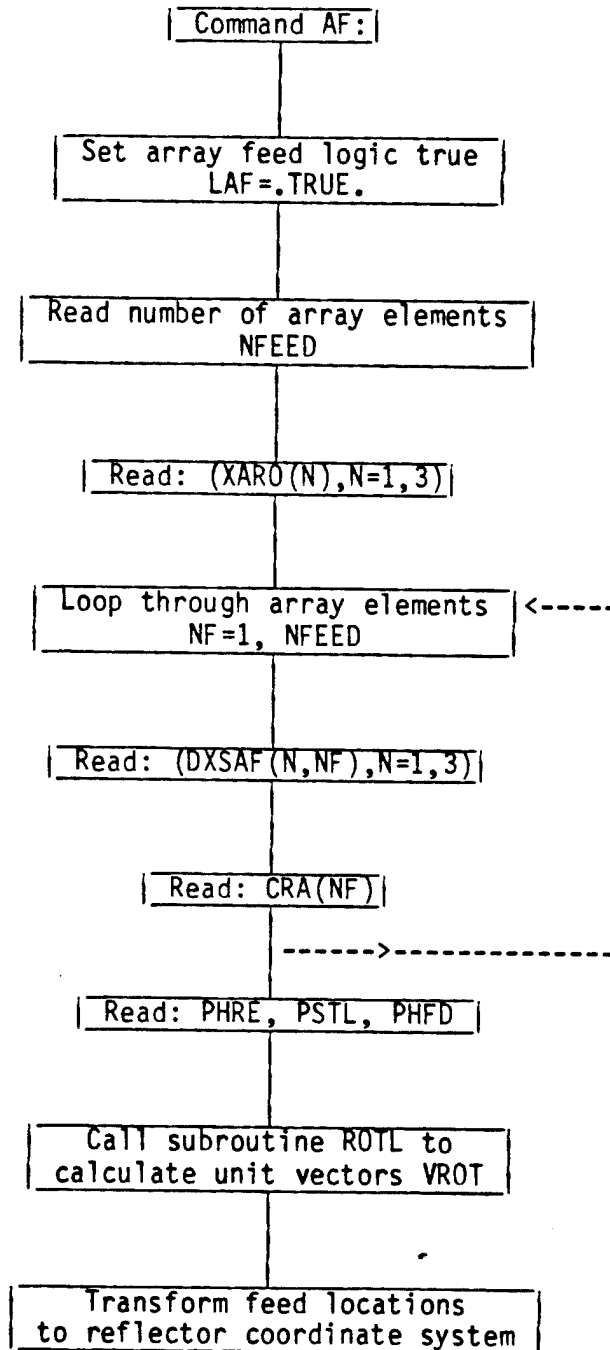
EXAMPLE:

This example demonstrates how the AF: command can be used for both method A and method B to calculate the reflector pattern with the array feed. A quadrant reflector as shown in Figure 4 with a 19-element array feed as shown in Figure 5 is used at the frequency of 15 GHz in this example. The elements for the array feed are $TE_{11}$ mode circular waveguides with diameter = 0.544 $\lambda$; and the pattern for a single element can be obtained by using the AP: command with inputs NMAG=11 and NPHA=1. The input data for this element pattern are given in Table 1.

In this example, the diameter of the array element is below the cutoff diameter for the $TE_{11}$ mode, thus CC(1)=1 so that the normalization constant for the $TE_{11}$ waveguide mode is ignored (see the AP: Command).

Table 1

INPUT DATA FOR CALCULATING ELEMENT PATTERN OF THE 19 ELEMENT ARRAY

```
CM:                  ***** TE11AF.DAT *****
CM:  CALCULATION OF CIRCULAR WAVEGUIDE TE11 MODE PATTERNS
CE:                  D=0.544 WAVELENGTHS
DG:
1
3    0.18   0.005   0.005   0.544   0
TO:
F    30.  1.
F    0    0    0    0
F .  F    F    F    0
T .  T    F    0.8
T    F    F    F    F    F    0.   0.
F    F    0.   0.
AP:
11   1.   0.   0.   0.   0.   0.        ! CC(1) = 1, IGNORE NORMAL. CONSTANT
 1   0.   0.   0.   0.   0.   0.
90.
FQ:
1    11.81102362
PZ:   LFDOUT=T
3
0.      45.      90.
0.      90.       1.
T
XQ:
```

After this run, the element pattern data are stored in the write unit #7 as shown in Table 2. It then should be edited to provide the FD: command of the input data for the next run.

# TABLE 2

## OUTPUT DATA FROM UNIT #7 FOR THE ELEMENT PATTERN

| | | | | |
|---|---|---|---|---|
| 0.000 | -6.074 | 90.000 | -87.617 | -89.990 |
| 1.000 | -6.075 | 90.000 | -87.619 | -89.990 |
| 2.000 | -6.079 | 90.000 | -87.627 | -89.990 |
| 3.000 | -6.085 | 90.000 | -87.640 | -89.990 |
| 4.000 | -6.094 | 90.000 | -87.657 | -89.990 |
| 5.000 | -6.106 | 90.000 | -87.681 | -89.990 |
| 6.000 | -6.120 | 90.000 | -87.709 | -89.990 |
| 7.000 | -6.137 | 90.000 | -87.742 | -89.990 |
| 8.000 | -6.156 | 90.000 | -87.780 | -89.991 |
| 9.000 | -6.177 | 90.000 | -87.823 | -89.990 |
| 10.000 | -6.201 | 90.000 | -87.871 | -89.990 |
| 11.000 | -6.228 | 90.000 | -87.925 | -89.990 |
| 12.000 | -6.257 | 90.000 | -87.983 | -89.991 |
| 13.000 | -6.289 | 90.000 | -88.046 | -89.991 |
| 14.000 | -6.323 | 90.000 | -88.115 | -89.991 |
| 15.000 | -6.359 | 90.000 | -88.188 | -89.991 |
| 16.000 | -6.398 | 90.000 | -88.266 | -89.991 |
| 17.000 | -6.440 | 90.000 | -88.349 | -89.991 |
| 18.000 | -6.484 | 90.000 | -88.437 | -89.991 |
| 19.000 | -6.530 | 90.000 | -88.529 | -89.991 |
| 20.000 | -6.578 | 90.000 | -88.627 | -89.991 |
| 21.000 | -6.629 | 90.000 | -88.729 | -89.991 |
| 22.000 | -6.682 | 90.000 | -88.836 | -89.991 |
| 23.000 | -6.737 | 90.000 | -88.948 | -89.992 |
| 24.000 | -6.795 | 90.000 | -89.064 | -89.992 |
| 25.000 | -6.855 | 90.000 | -89.185 | -89.992 |
| 26.000 | -6.917 | 90.000 | -89.310 | -89.992 |
| 27.000 | -6.981 | 90.000 | -89.440 | -89.992 |
| 28.000 | -7.048 | 90.000 | -89.574 | -89.992 |
| 29.000 | -7.117 | 90.000 | -89.713 | -89.992 |
| 30.000 | -7.187 | 90.000 | -89.856 | -89.992 |
| 31.000 | -7.260 | 90.000 | -90.004 | -89.992 |
| 32.000 | -7.335 | 90.000 | -90.155 | -89.992 |
| 33.000 | -7.412 | 90.000 | -90.311 | -89.992 |
| 34.000 | -7.490 | 90.000 | -90.471 | -89.992 |
| 35.000 | -7.571 | 90.000 | -90.634 | -89.992 |
| 36.000 | -7.654 | 90.000 | -90.802 | -89.992 |
| 37.000 | -7.738 | 90.000 | -90.973 | -89.993 |
| 38.000 | -7.825 | 90.000 | -91.149 | -89.993 |
| 39.000 | -7.913 | 90.000 | -91.328 | -89.993 |
| 40.000 | -8.003 | 90.000 | -91.510 | -89.993 |
| 41.000 | -8.095 | 90.000 | -91.696 | -89.993 |
| 42.000 | -8.188 | 90.000 | -91.885 | -89.993 |
| 43.000 | -8.283 | 90.000 | -92.078 | -89.993 |
| 44.000 | -8.380 | 90.000 | -92.274 | -89.993 |
| 45.000 | -8.479 | 90.000 | -92.473 | -89.993 |
| 46.000 | -8.579 | 90.000 | -92.674 | -89.993 |
| 47.000 | -8.680 | 90.000 | -92.879 | -89.993 |
| 48.000 | -8.783 | 90.000 | -93.086 | -89.993 |
| 49.000 | -8.888 | 90.000 | -93.296 | -89.994 |
| 50.000 | -8.994 | 90.000 | -93.508 | -89.994 |
| 51.000 | -9.102 | 90.000 | -93.722 | -89.994 |
| 52.000 | -9.210 | 90.000 | -93.939 | -89.994 |
| 53.000 | -9.321 | 90.000 | -94.157 | -89.994 |
| 54.000 | -9.432 | 90.000 | -94.377 | -89.994 |
| 55.000 | -9.545 | 90.000 | -94.599 | -89.994 |
| 56.000 | -9.659 | 90.000 | -94.822 | -89.994 |
| 57.000 | -9.775 | 90.000 | -95.047 | -89.994 |
| 58.000 | -9.891 | 90.000 | -95.272 | -89.994 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 59.000 | −10.009 | 90.000 | −95.499 | −89.995 |
| 60.000 | −10.128 | 90.000 | −95.726 | −89.995 |
| 61.000 | −10.248 | 90.000 | −95.953 | −89.995 |
| 62.000 | −10.370 | 90.000 | −96.181 | −89.995 |
| 63.000 | −10.492 | 90.000 | −96.409 | −89.995 |
| 64.000 | −10.615 | 90.000 | −96.636 | −89.995 |
| 65.000 | −10.740 | 90.000 | −96.864 | −89.995 |
| 66.000 | −10.865 | 90.000 | −97.090 | −89.995 |
| 67.000 | −10.992 | 90.000 | −97.316 | −89.995 |
| 68.000 | −11.120 | 90.000 | −97.541 | −89.996 |
| 69.000 | −11.248 | 90.000 | −97.764 | −89.995 |
| 70.000 | −11.378 | 90.000 | −97.986 | −89.996 |
| 71.000 | −11.509 | 90.000 | −98.207 | −89.996 |
| 72.000 | −11.640 | 90.000 | −98.425 | −89.996 |
| 73.000 | −11.773 | 90.000 | −98.641 | −89.996 |
| 74.000 | −11.906 | 90.000 | −98.854 | −89.996 |
| 75.000 | −12.041 | 90.000 | −99.066 | −89.996 |
| 76.000 | −12.176 | 90.000 | −99.274 | −89.996 |
| 77.000 | −12.313 | 90.000 | −99.479 | −89.996 |
| 78.000 | −12.450 | 90.000 | −99.681 | −89.996 |
| 79.000 | −12.589 | 90.000 | −99.880 | −89.996 |
| 80.000 | −12.729 | 90.000 | −100.075 | −89.996 |
| 81.000 | −12.869 | 90.000 | −100.266 | −89.996 |
| 82.000 | −13.011 | 90.000 | −100.453 | −89.997 |
| 83.000 | −13.153 | 90.000 | −100.637 | −89.997 |
| 84.000 | −13.297 | 90.000 | −100.816 | −89.997 |
| 85.000 | −13.442 | 90.000 | −100.991 | −89.997 |
| 86.000 | −13.588 | 90.000 | −101.162 | −89.997 |
| 87.000 | −13.735 | 90.000 | −101.329 | −89.997 |
| 88.000 | −13.884 | 90.000 | −101.492 | −89.997 |
| 89.000 | −14.033 | 90.000 | −101.650 | −89.997 |
| 90.000 | −300.000 | 0.000 | −300.000 | 0.000 |
| 0.000 | −6.075 | 90.000 | −99.204 | −87.982 |
| 1.000 | −6.076 | 90.000 | −119.555 | −77.638 |
| 2.000 | −6.080 | 90.000 | −90.861 | 89.870 |
| 3.000 | −6.088 | 90.000 | −82.151 | 90.070 |
| 4.000 | −6.098 | 90.000 | −76.650 | 90.100 |
| 5.000 | −6.111 | 90.000 | −72.564 | 90.101 |
| 6.000 | −6.127 | 90.000 | −69.332 | 90.098 |
| 7.000 | −6.146 | 90.000 | −66.503 | 90.090 |
| 8.000 | −6.167 | 90.000 | −64.241 | 90.083 |
| 9.000 | −6.192 | 90.000 | −62.165 | 90.077 |
| 10.000 | −6.219 | 90.000 | −60.382 | 90.072 |
| 11.000 | −6.250 | 90.000 | −58.760 | 90.067 |
| 12.000 | −6.283 | 90.000 | −57.269 | 90.063 |
| 13.000 | −6.319 | 90.000 | −55.910 | 90.059 |
| 14.000 | −6.357 | 90.000 | −54.678 | 90.056 |
| 15.000 | −6.398 | 90.000 | −53.530 | 90.053 |
| 16.000 | −6.443 | 90.000 | −52.468 | 90.051 |
| 17.000 | −6.489 | 90.000 | −51.476 | 90.048 |
| 18.000 | −6.539 | 90.000 | −50.548 | 90.046 |
| 19.000 | −6.591 | 90.000 | −49.680 | 90.044 |
| 20.000 | −6.646 | 90.000 | −48.864 | 90.042 |
| 21.000 | −6.704 | 90.000 | −48.097 | 90.041 |
| 22.000 | −6.764 | 90.000 | −47.373 | 90.039 |
| 23.000 | −6.826 | 90.000 | −46.688 | 90.038 |
| 24.000 | −6.891 | 90.000 | −46.040 | 90.036 |
| 25.000 | −6.959 | 90.000 | −45.427 | 90.035 |
| 26.000 | −7.029 | 90.000 | −44.846 | 90.034 |
| 27.000 | −7.102 | 90.000 | −44.295 | 90.033 |
| 28.000 | −7.177 | 90.000 | −43.771 | 90.032 |
| 29.000 | −7.254 | 90.000 | −43.273 | 90.031 |
| 30.000 | −7.333 | 90.000 | −42.800 | 90.030 |
| 31.000 | −7.415 | 90.000 | −42.350 | 90.030 |
| 32.000 | −7.499 | 90.000 | −41.921 | 90.029 |
| 33.000 | −7.586 | 90.000 | −41.514 | 90.028 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 34.000 | -7.674 | 90.000 | -41.126 | 90.028 |
| 35.000 | -7.765 | 90.000 | -40.757 | 90.027 |
| 36.000 | -7.857 | 90.000 | -40.406 | 90.026 |
| 37.000 | -7.952 | 90.000 | -40.073 | 90.026 |
| 38.000 | -8.049 | 90.000 | -39.755 | 90.025 |
| 39.000 | -8.147 | 90.000 | -39.454 | 90.025 |
| 40.000 | -8.248 | 90.000 | -39.167 | 90.024 |
| 41.000 | -8.350 | 90.000 | -38.895 | 90.024 |
| 42.000 | -8.454 | 90.000 | -38.637 | 90.024 |
| 43.000 | -8.560 | 90.000 | -38.392 | 90.023 |
| 44.000 | -8.668 | 90.000 | -38.161 | 90.023 |
| 45.000 | -8.777 | 90.000 | -37.941 | 90.022 |
| 46.000 | -8.888 | 90.000 | -37.735 | 90.022 |
| 47.000 | -9.000 | 90.000 | -37.539 | 90.022 |
| 48.000 | -9.115 | 90.000 | -37.356 | 90.021 |
| 49.000 | -9.230 | 90.000 | -37.183 | 90.021 |
| 50.000 | -9.347 | 90.000 | -37.021 | 90.021 |
| 51.000 | -9.465 | 90.000 | -36.870 | 90.021 |
| 52.000 | -9.585 | 90.000 | -36.729 | 90.020 |
| 53.000 | -9.706 | 90.000 | -36.598 | 90.020 |
| 54.000 | -9.828 | 90.000 | -36.477 | 90.020 |
| 55.000 | -9.952 | 90.000 | -36.365 | 90.020 |
| 56.000 | -10.076 | 90.000 | -36.262 | 90.020 |
| 57.000 | -10.202 | 90.000 | -36.168 | 90.019 |
| 58.000 | -10.329 | 90.000 | -36.083 | 90.019 |
| 59.000 | -10.457 | 90.000 | -36.007 | 90.019 |
| 60.000 | -10.586 | 90.000 | -35.940 | 90.019 |
| 61.000 | -10.716 | 90.000 | -35.881 | 90.019 |
| 62.000 | -10.847 | 90.000 | -35.830 | 90.018 |
| 63.000 | -10.979 | 90.000 | -35.787 | 90.018 |
| 64.000 | -11.111 | 90.000 | -35.752 | 90.018 |
| 65.000 | -11.245 | 90.000 | -35.724 | 90.018 |
| 66.000 | -11.379 | 90.000 | -35.705 | 90.018 |
| 67.000 | -11.514 | 90.000 | -35.693 | 90.018 |
| 68.000 | -11.650 | 90.000 | -35.688 | 90.018 |
| 69.000 | -11.787 | 90.000 | -35.691 | 90.018 |
| 70.000 | -11.924 | 90.000 | -35.701 | 90.018 |
| 71.000 | -12.062 | 90.000 | -35.719 | 90.017 |
| 72.000 | -12.201 | 90.000 | -35.743 | 90.017 |
| 73.000 | -12.340 | 90.000 | -35.775 | 90.017 |
| 74.000 | -12.480 | 90.000 | -35.814 | 90.017 |
| 75.000 | -12.621 | 90.000 | -35.859 | 90.017 |
| 76.000 | -12.762 | 90.000 | -35.912 | 90.017 |
| 77.000 | -12.904 | 90.000 | -35.971 | 90.017 |
| 78.000 | -13.046 | 90.000 | -36.038 | 90.017 |
| 79.000 | -13.190 | 90.000 | -36.111 | 90.017 |
| 80.000 | -13.333 | 90.000 | -36.191 | 90.017 |
| 81.000 | -13.478 | 90.000 | -36.278 | 90.017 |
| 82.000 | -13.623 | 90.000 | -36.371 | 90.017 |
| 83.000 | -13.769 | 90.000 | -36.472 | 90.017 |
| 84.000 | -13.915 | 90.000 | -36.579 | 90.017 |
| 85.000 | -14.062 | 90.000 | -36.693 | 90.017 |
| 86.000 | -14.210 | 90.000 | -36.813 | 90.017 |
| 87.000 | -14.359 | 90.000 | -36.941 | 90.017 |
| 88.000 | -14.508 | 90.000 | -37.075 | 90.017 |
| 89.000 | -14.659 | 90.000 | -37.216 | 90.017 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | -6.077 | 90.000 | -90.435 | -89.986 |
| 1.000 | -6.078 | 90.000 | -90.437 | -90.408 |
| 2.000 | -6.083 | 90.000 | -90.444 | -90.830 |
| 3.000 | -6.091 | 90.000 | -90.456 | -91.253 |
| 4.000 | -6.103 | 90.000 | -90.472 | -91.676 |
| 5.000 | -6.117 | 90.000 | -90.493 | -92.099 |
| 6.000 | -6.135 | 90.000 | -90.519 | -92.524 |
| 7.000 | -6.156 | 90.000 | -90.549 | -92.949 |
| 8.000 | -6.181 | 90.000 | -90.584 | -93.376 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 9.000 | -6.208 | 90.000 | -90.624 | -93.804 |
| 10.000 | -6.239 | 90.000 | -90.668 | -94.234 |
| 11.000 | -6.272 | 90.000 | -90.717 | -94.664 |
| 12.000 | -6.309 | 90.000 | -90.770 | -95.098 |
| 13.000 | -6.350 | 90.000 | -90.828 | -95.533 |
| 14.000 | -6.393 | 90.000 | -90.890 | -95.970 |
| 15.000 | -6.439 | 90.000 | -90.957 | -96.409 |
| 16.000 | -6.488 | 90.000 | -91.028 | -96.851 |
| 17.000 | -6.541 | 90.000 | -91.104 | -97.296 |
| 18.000 | -6.596 | 90.000 | -91.184 | -97.744 |
| 19.000 | -6.654 | 90.000 | -91.268 | -98.194 |
| 20.000 | -6.716 | 90.000 | -91.356 | -98.647 |
| 21.000 | -6.780 | 90.000 | -91.449 | -99.104 |
| 22.000 | -6.847 | 90.000 | -91.546 | -99.564 |
| 23.000 | -6.917 | 90.000 | -91.646 | -100.027 |
| 24.000 | -6.990 | 90.000 | -91.751 | -100.494 |
| 25.000 | -7.065 | 90.000 | -91.860 | -100.965 |
| 26.000 | -7.144 | 90.000 | -91.973 | -101.439 |
| 27.000 | -7.224 | 90.000 | -92.089 | -101.917 |
| 28.000 | -7.308 | 90.000 | -92.209 | -102.400 |
| 29.000 | -7.394 | 90.000 | -92.333 | -102.886 |
| 30.000 | -7.483 | 90.000 | -92.461 | -103.377 |
| 31.000 | -7.574 | 90.000 | -92.592 | -103.871 |
| 32.000 | -7.668 | 90.000 | -92.726 | -104.370 |
| 33.000 | -7.764 | 90.000 | -92.864 | -104.874 |
| 34.000 | -7.863 | 90.000 | -93.005 | -105.382 |
| 35.000 | -7.963 | 90.000 | -93.149 | -105.894 |
| 36.000 | -8.066 | 90.000 | -93.296 | -106.410 |
| 37.000 | -8.172 | 90.000 | -93.446 | -106.930 |
| 38.000 | -8.279 | 90.000 | -93.599 | -107.455 |
| 39.000 | -8.389 | 90.000 | -93.755 | -107.983 |
| 40.000 | -8.500 | 90.000 | -93.913 | -108.516 |
| 41.000 | -8.614 | 90.000 | -94.074 | -109.053 |
| 42.000 | -8.729 | 90.000 | -94.237 | -109.594 |
| 43.000 | -8.846 | 90.000 | -94.403 | -110.138 |
| 44.000 | -8.966 | 90.000 | -94.570 | -110.685 |
| 45.000 | -9.086 | 90.000 | -94.740 | -111.236 |
| 46.000 | -9.209 | 90.000 | -94.911 | -111.790 |
| 47.000 | -9.333 | 90.000 | -95.085 | -112.346 |
| 48.000 | -9.459 | 90.000 | -95.260 | -112.904 |
| 49.000 | -9.586 | 90.000 | -95.436 | -113.465 |
| 50.000 | -9.715 | 90.000 | -95.614 | -114.027 |
| 51.000 | -9.845 | 90.000 | -95.793 | -114.590 |
| 52.000 | -9.977 | 90.000 | -95.973 | -115.153 |
| 53.000 | -10.109 | 90.000 | -96.154 | -115.717 |
| 54.000 | -10.243 | 90.000 | -96.336 | -116.280 |
| 55.000 | -10.379 | 90.000 | -96.519 | -116.842 |
| 56.000 | -10.515 | 90.000 | -96.702 | -117.402 |
| 57.000 | -10.652 | 90.000 | -96.885 | -117.961 |
| 58.000 | -10.790 | 90.000 | -97.069 | -118.515 |
| 59.000 | -10.929 | 90.000 | -97.253 | -119.067 |
| 60.000 | -11.069 | 90.000 | -97.437 | -119.613 |
| 61.000 | -11.210 | 90.000 | -97.621 | -120.155 |
| 62.000 | -11.352 | 90.000 | -97.805 | -120.690 |
| 63.000 | -11.494 | 90.000 | -97.989 | -121.218 |
| 64.000 | -11.637 | 90.000 | -98.172 | -121.738 |
| 65.000 | -11.781 | 90.000 | -98.354 | -122.248 |
| 66.000 | -11.925 | 90.000 | -98.536 | -122.750 |
| 67.000 | -12.070 | 90.000 | -98.717 | -123.240 |
| 68.000 | -12.215 | 90.000 | -98.898 | -123.719 |
| 69.000 | -12.360 | 90.000 | -99.077 | -124.185 |
| 70.000 | -12.506 | 90.000 | -99.256 | -124.638 |
| 71.000 | -12.653 | 90.000 | -99.433 | -125.076 |
| 72.000 | -12.800 | 90.000 | -99.610 | -125.498 |
| 73.000 | -12.947 | 90.000 | -99.785 | -125.903 |
| 74.000 | -13.094 | 90.000 | -99.959 | -126.292 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 75.000 | -13.242 | 90.000 | -100.131 | -126.662 |
| 76.000 | -13.389 | 90.000 | -100.303 | -127.012 |
| 77.000 | -13.538 | 90.000 | -100.473 | -127.343 |
| 78.000 | -13.686 | 90.000 | -100.641 | -127.652 |
| 79.000 | -13.834 | 90.000 | -100.808 | -127.940 |
| 80.000 | -13.983 | 90.000 | -100.974 | -128.206 |
| 81.000 | -14.132 | 90.000 | -101.139 | -128.448 |
| 82.000 | -14.281 | 90.000 | -101.302 | -128.667 |
| 83.000 | -14.430 | 90.000 | -101.463 | -128.861 |
| 84.000 | -14.580 | 90.000 | -101.623 | -129.031 |
| 85.000 | -14.730 | 90.000 | -101.782 | -129.175 |
| 86.000 | -14.880 | 90.000 | -101.939 | -129.293 |
| 87.000 | -15.030 | 90.000 | -102.095 | -129.386 |
| 88.000 | -15.181 | 90.000 | -102.250 | -129.452 |
| 89.000 | -15.332 | 90.000 | -102.404 | -129.492 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
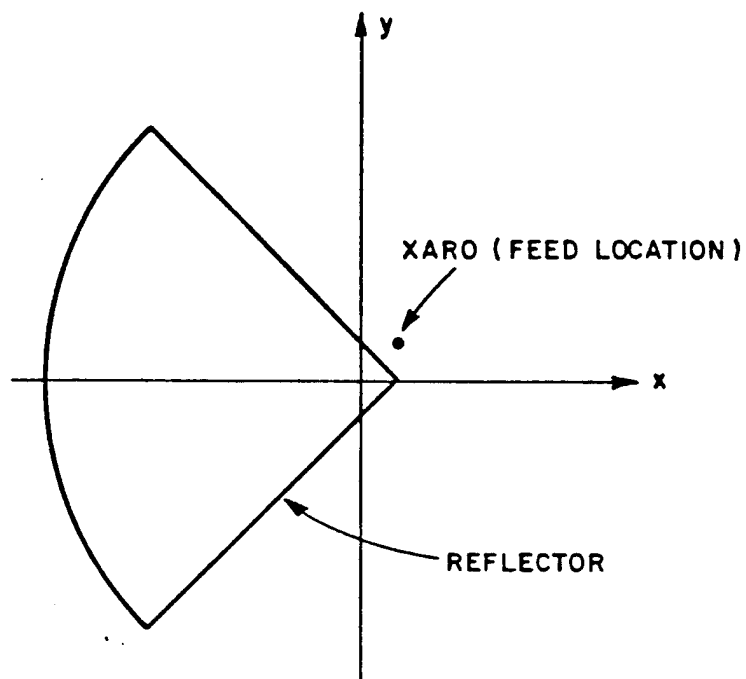
Figure 4. Front view of the quadrant reflector antenna.



$D = 0.544 \lambda$
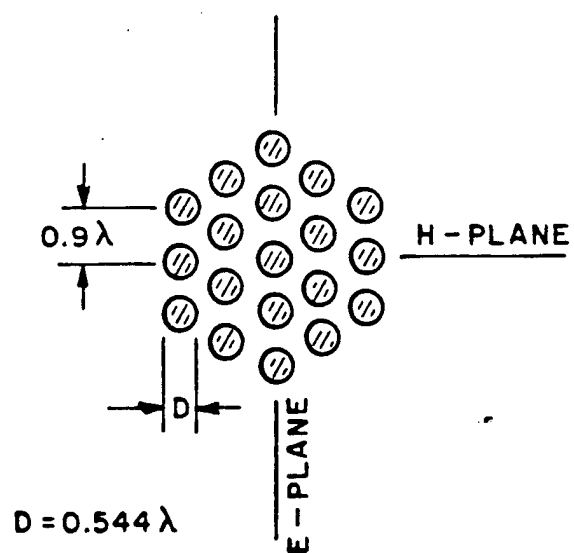
Figure 5. 19-element array feed.

1.  Method A

    The reflector pattern is calculated directly in this run.  The
input data are given in Table 3.  Note that part of the FD:  command is
not listed since it comes from the data file FOR007.DAT as shown in
Table 2.

## TABLE 3

### INPUT DATA FOR CALCULATING THE QUADRANT REFLECTOR PATTERN
### (METHOD A)

```
CM:        ***** RF19X.DAT *****
CM:        EXAMPLE OF ARRAY FEED
CM:        --- TILT= 21.6 DEG. ---
CM:        --- AIC ONLY ---
CM:        ARRAY FEED, NFEED=19
CE: CALCULATE REFLECTOR PATTERN DIRECTLY
DG:        EDGES ROTATED 90 DEGREES
1
3 . 137.7    3.   3.   0    22
  -70.22    78.02
  -74.58    74.58
  -80.79    67.79
  -86.39    60.49
  -91.34    52.73
  -95.58    44.57
  -99.11    36.07
 -101.87    27.30
 -103.86    18.31
 -105.06     9.19
 -105.47     0.00
 -105.06    -9.19
 -103.86   -18.31
 -101.87   -27.30
  -99.11   -36.07
  -95.58   -44.57
  -91.34   -52.73
  -86.39   -60.49
  -80.79   -67.79
  -74.58   -74.58
  -70.22   -78.02
    7.80     0.00
TO: LAIC = T ONLY
F  180.   5.
F   27   27   1    55
F    F    F    F    0
T    T    F    0.8
T    F    F    F    F    F    F   0.   0.
F    F    0.   0.
FD: TE11 CIRCULAR WAVEGUIDE FOR SINGLE ELEMENT
0    T
T    0    T    1    90.  0.  F
3    0.  45.  90.
91
      0.000      -6.074     90.000     -87.617     -89.990
      1.000      -6.075     90.000     -87.619     -89.990
      2.000      -6.079     90.000     -87.627     -89.990
      3.000      -6.085     90.000     -87.640     -89.990
      4.000      -6.094     90.000     -87.657     -89.990
        .            .          .           .           .
        .            .          .           .           .
        .            .          .           .           .

                    ( see Table 2. )

        .            .          .           .           .
        .            .          .           .           .
        .            .          .           .           .
     86.000     -14.880     90.000    -101.939    -129.293
     87.000     -15.030     90.000    -102.095    -129.386
```

TABLE 3 - CONTINUED

```
      88.000      -15.181      90.000     -102.250    -129.452
      89.000      -15.332      90.000     -102.404    -129.492
      90.000     -300.000       0.000     -300.000       0.000
AF: 19 ELEMENTS ARRAY FEED
19
7.8    7.8    137.7
0.         0.          0.
1.     0.
0.         -0.7087     0.
0.75   0.
0.6138  -0.3543     0.
0.75   0.
0.6138   0.3543     0.
0.75   0.
0.         0.7087     0.
0.75   0.
-0.6138   0.3543     0.
0.75   0.
-0.6138  -0.3543     0.
0.75   0.
0.         -1.4173     0.
0.25   0.
0.6138  -1.0630     0.
0.25   0.
1.2275  -0.7087     0.
0.25   0.
1.2275   0.          0.
0.25   0.
1.2275   0.7087     0.
0.25   0.
0.6138   1.0630     0.
0.25   0.
0.         1.4173     0.
0.25   0.
-0.6138   1.0630     0.
0.25   0.
-1.2275   0.7087     0.
0.25   0.
-1.2275   0.          0.
0.25   0.
-1.2275  -0.7087     0.
0.25   0.
-0.6138  -1.0630     0.
0.25   0.
90.    21.6    90.
FQ:
1    15.
PZ:
1
-135
-12.     12.      0.05
F
PP:
1
1   1
1   2
XQ:
```

The reflector pattern for the φ=-135° cut is shown in Figure 6. Note that the gain level is incorrect because a single element is used for the reference gain. Also note that with this method the total aperture fields are calculated by superimposing the aperture field of each array element. This is not necessary for this case because the reflector is in the far field of the feed.

PHI (DEG) = -135.0
F    (WL) = 174.9
MAX      = 28.76
AIC
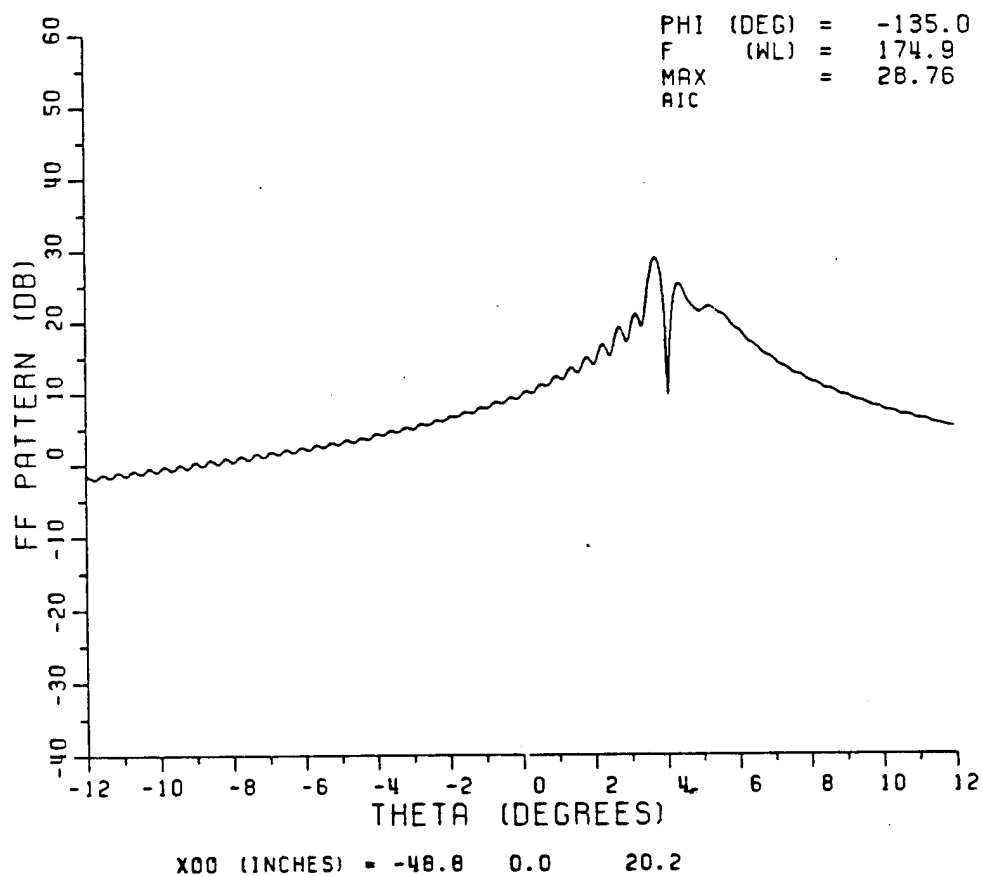
Figure 6.  The reflector pattern for the φ = -135° cut by using method A.

2. Method B

    1) Step 1:  The feed pattern of the array is calculated and stored in write unit #7 by using the input data given in Table 4.

## TABLE 4

### INPUT DATA FOR CALCULATING ARRAY PATTERN
### (METHOD B)

```
CM:    ***** RF19F2.DAT *****
CM:      EXAMPLE OF ARRAY FEED
CM:    CALCULATE THE ARRAY PATTERN
CE:      ARRAY FEED, NFEED=19
DG:    NOT AFFECTED IN THIS RUN
1
3    8.   12.   12.   24.   0
TO:
F    0.   0.
F .  0    0    0    0
F    F    F    F    0
T    T    F    0.8
T    F    F    F    F    F    0.   0.
F    F    0.   0.
FD:  TE11 CIRCULAR WAVEGUIDE FOR SINGLE ELEMENT
0    T .
T    0    T    1    90.   0.   F
3    0.   45.   90.
91
      0.000      -6.074      90.000      -87.617      -89.990
      1.000      -6.075      90.000      -87.619      -89.990
      2.000      -6.079      90.000      -87.627      -89.990
      3.000      -6.085      90.000      -87.640      -89.990
      4.000      -6.094      90.000      -87.657      -89.990
         .           .           .           .           .
         .           .           .           .           .
         .           .           .           .           .

                      ( see Table 2. )

         .           .           .           .           .
         .           .           .           .           .
         .           .           .           .           .
     86.000     -14.880      90.000     -101.939     -129.293
     87.000     -15.030      90.000     -102.095     -129.386
     88.000     -15.181      90.000     -102.250     -129.452
     89.000     -15.332      90.000     -102.404     -129.492
     90.000    -300.000       0.000     -300.000        0.000
AF: 19 ELEMENTS ARRAY FEED
19
0.  0.  137.7
0.          0.          0.
1.    0.
0.        -0.7087     0.
0.75  0.
0.6138  -0.3543     0.
0.75  0.
0.6138    0.3543     0.
0.75  0.
0.         0.7087     0.
0.75  0.
-0.6138    0.3543     0.
0.75  0.
-0.6138  -0.3543     0.
0.75  0.
0.        -1.4173     0.
0.25  0.
0.6138  -1.0630     0.
0.25  0.
1.2275  -0.7087     0.
0.25  0.
```

TABLE 4 - CONTINUED

```
1.2275    0.          0.
0.25  0.
1.2275    0.7087      0.
0.25  0.
0.6138    1.0630      0.
0.25  0.
0.        1.4173      0.
0.25  0.
-0.6138   1.0630      0.
0.25  0.
-1.2275   0.7087      0.
0.25  0.
-1.2275   0.          0.
0.25  0.
-1.2275   -0.7087     0.
0.25  0.
-0.6138   -1.0630     0.
0.25  0.
90.   21.6   90.
FQ:
1     15.
PF:
10
0. 10. 20. 30. 40. 50. 60. 70. 80. 90.
0. 90. 1.
1
2
PZ: LFDOUT=T
1
-135.
12.      12.      0.05
T
XQ:
```

The output data in the unit #7 as shown in Table 5 are then edited for the input data for the FD:  command in step 2.

# TABLE 5

## OUTPUT DATA FROM UNIT #7 FOR THE ARRAY PATTERN

| | | | | |
|---|---|---|---|---|
| 0.000 | 12.514 | 90.000 | -300.000 | 0.000 |
| 1.000 | 12.476 | 90.000 | -300.000 | 0.000 |
| 2.000 | 12.360 | 90.000 | -300.000 | 0.000 |
| 3.000 | 12.166 | 90.000 | -300.000 | 0.000 |
| 4.000 | 11.894 | 90.000 | -300.000 | 0.000 |
| 5.000 | 11.541 | 90.000 | -300.000 | 0.000 |
| 6.000 | 11.107 | 90.000 | -300.000 | 0.000 |
| 7.000 | 10.590 | 90.000 | -300.000 | 0.000 |
| 8.000 | 9.988 | 90.000 | -300.000 | 0.000 |
| 9.000 | 9.299 | 90.000 | -300.000 | 0.000 |
| 10.000 | 8.517 | 90.000 | -300.000 | 0.000 |
| 11.000 | 7.640 | 90.000 | -300.000 | 0.000 |
| 12.000 | 6.662 | 90.000 | -300.000 | 0.000 |
| 13.000 | 5.579 | 90.000 | -300.000 | 0.000 |
| 14.000 | 4.385 | 90.000 | -300.000 | 0.000 |
| 15.000 | 3.071 | 90.000 | -300.000 | 0.000 |
| 16.000 | 1.629 | 90.000 | -300.000 | 0.000 |
| 17.000 | 0.049 | 90.000 | -300.000 | 0.000 |
| 18.000 | -1.679 | 90.000 | -300.000 | 0.000 |
| 19.000 | -3.567 | 90.000 | -300.000 | 0.000 |
| 20.000 | -5.627 | 90.000 | -300.000 | 0.000 |
| 21.000 | -7.867 | 90.000 | -300.000 | 0.000 |
| 22.000 | -10.287 | 90.000 | -300.000 | 0.000 |
| 23.000 | -12.865 | 90.000 | -300.000 | 0.000 |
| 24.000 | -15.538 | 90.000 | -300.000 | 0.000 |
| 25.000 | -18.155 | 90.000 | -300.000 | 0.000 |
| 26.000 | -20.446 | 90.000 | -300.000 | 0.000 |
| 27.000 | -22.038 | 90.000 | -300.000 | 0.000 |
| 28.000 | -22.640 | 90.000 | -300.000 | 0.000 |
| 29.000 | -22.276 | 90.000 | -300.000 | 0.000 |
| 30.000 | -21.261 | 90.000 | -300.000 | 0.000 |
| 31.000 | -19.964 | 90.000 | -300.000 | 0.000 |
| 32.000 | -18.643 | 90.000 | -300.000 | 0.000 |
| 33.000 | -17.434 | 90.000 | -300.000 | 0.000 |
| 34.000 | -16.395 | 90.000 | -300.000 | 0.000 |
| 35.000 | -15.548 | 90.000 | -300.000 | 0.000 |
| 36.000 | -14.893 | 90.000 | -300.000 | 0.000 |
| 37.000 | -14.421 | 90.000 | -300.000 | 0.000 |
| 38.000 | -14.126 | 90.000 | -300.000 | 0.000 |
| 39.000 | -13.996 | 90.000 | -300.000 | 0.000 |
| 40.000 | -14.024 | 90.000 | -300.000 | 0.000 |
| 41.000 | -14.203 | 90.000 | -300.000 | 0.000 |
| 42.000 | -14.526 | 90.000 | -300.000 | 0.000 |
| 43.000 | -14.988 | 90.000 | -300.000 | 0.000 |
| 44.000 | -15.586 | 90.000 | -300.000 | 0.000 |
| 45.000 | -16.313 | 90.000 | -300.000 | 0.000 |
| 46.000 | -17.163 | 90.000 | -300.000 | 0.000 |
| 47.000 | -18.125 | 90.000 | -300.000 | 0.000 |
| 48.000 | -19.185 | 90.000 | -300.000 | 0.000 |
| 49.000 | -20.317 | 90.000 | -300.000 | 0.000 |
| 50.000 | -21.480 | 90.000 | -300.000 | 0.000 |
| 51.000 | -22.615 | 90.000 | -300.000 | 0.000 |
| 52.000 | -23.634 | 90.000 | -300.000 | 0.000 |
| 53.000 | -24.436 | 90.000 | -300.000 | 0.000 |
| 54.000 | -24.910 | 90.000 | -300.000 | 0.000 |
| 55.000 | -24.981 | 90.000 | -300.000 | 0.000 |
| 56.000 | -24.630 | 90.000 | -300.000 | 0.000 |
| 57.000 | -23.907 | 90.000 | -300.000 | 0.000 |
| 58.000 | -22.904 | 90.000 | -300.000 | 0.000 |
| 59.000 | -21.731 | 90.000 | -300.000 | 0.000 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 60.000 | -20.475 | 90.000 | -300.000 | 0.000 |
| 61.000 | -19.205 | 90.000 | -300.000 | 0.000 |
| 62.000 | -17.964 | 90.000 | -300.000 | 0.000 |
| 63.000 | -16.779 | 90.000 | -300.000 | 0.000 |
| 64.000 | -15.664 | 90.000 | -300.000 | 0.000 |
| 65.000 | -14.626 | 90.000 | -300.000 | 0.000 |
| 66.000 | -13.666 | 90.000 | -300.000 | 0.000 |
| 67.000 | -12.783 | 90.000 | -300.000 | 0.000 |
| 68.000 | -11.976 | 90.000 | -300.000 | 0.000 |
| 69.000 | -11.238 | 90.000 | -300.000 | 0.000 |
| 70.000 | -10.568 | 90.000 | -300.000 | 0.000 |
| 71.000 | -9.961 | 90.000 | -300.000 | 0.000 |
| 72.000 | -9.413 | 90.000 | -300.000 | 0.000 |
| 73.000 | -8.922 | 90.000 | -300.000 | 0.000 |
| 74.000 | -8.482 | 90.000 | -300.000 | 0.000 |
| 75.000 | -8.093 | 90.000 | -300.000 | 0.000 |
| 76.000 | -7.749 | 90.000 | -300.000 | 0.000 |
| 77.000 | -7.450 | 90.000 | -300.000 | 0.000 |
| 78.000 | -7.192 | 90.000 | -300.000 | 0.000 |
| 79.000 | -6.975 | 90.000 | -300.000 | 0.000 |
| 80.000 | -6.795 | 90.000 | -300.000 | 0.000 |
| 81.000 | -6.651 | 90.000 | -300.000 | 0.000 |
| 82.000 | -6.542 | 90.000 | -300.000 | 0.000 |
| 83.000 | -6.466 | 90.000 | -300.000 | 0.000 |
| 84.000 | -6.422 | 90.000 | -300.000 | 0.000 |
| 85.000 | -6.411 | 90.000 | -300.000 | 0.000 |
| 86.000 | -6.429 | 90.000 | -300.000 | 0.000 |
| 87.000 | -6.478 | 90.000 | -300.000 | 0.000 |
| 88.000 | -6.557 | 90.000 | -300.000 | 0.000 |
| 89.000 | -6.665 | 90.000 | -300.000 | 0.000 |
| 90.000 | -292.618 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.514 | 90.000 | -71.604 | -89.544 |
| 1.000 | 12.476 | 90.000 | -76.165 | -87.245 |
| 2.000 | 12.360 | 90.000 | -69.907 | -50.021 |
| 3.000 | 12.166 | 90.000 | -68.169 | -129.977 |
| 4.000 | 11.893 | 90.000 | -67.223 | -129.970 |
| 5.000 | 11.540 | 90.000 | -66.675 | -129.970 |
| 6.000 | 11.106 | 90.000 | -66.398 | -129.970 |
| 7.000 | 10.588 | 90.000 | -66.295 | -129.972 |
| 8.000 | 9.986 | 90.000 | -66.405 | -129.975 |
| 9.000 | 9.295 | 90.000 | -66.646 | -129.975 |
| 10.000 | 8.511 | 90.000 | -67.046 | -129.976 |
| 11.000 | 7.632 | 90.000 | -67.579 | -129.977 |
| 12.000 | 6.651 | 90.000 | -68.244 | -129.979 |
| 13.000 | 5.563 | 90.000 | -69.046 | -129.980 |
| 14.000 | 4.361 | 90.000 | -69.993 | -129.981 |
| 15.000 | 3.036 | 90.000 | -71.082 | -129.981 |
| 16.000 | 1.576 | 90.000 | -72.327 | -129.982 |
| 17.000 | -0.032 | 90.000 | -73.736 | -129.982 |
| 18.000 | -1.804 | 90.000 | -75.325 | -129.983 |
| 19.000 | -3.762 | 90.000 | -77.114 | -129.983 |
| 20.000 | -5.932 | 90.000 | -79.130 | -129.984 |
| 21.000 | -8.352 | 90.000 | -81.406 | -129.984 |
| 22.000 | -11.069 | 90.000 | -83.991 | -129.984 |
| 23.000 | -14.147 | 90.000 | -86.947 | -129.985 |
| 24.000 | -17.674 | 90.000 | -90.361 | -129.986 |
| 25.000 | -21.768 | 90.000 | -94.351 | -129.986 |
| 26.000 | -26.556 | 90.000 | -99.043 | -129.986 |
| 27.000 | -32.040 | 90.000 | -104.439 | -129.987 |
| 28.000 | -37.378 | 90.000 | -109.697 | -129.987 |
| 29.000 | -39.553 | 90.000 | -111.798 | -129.987 |
| 30.000 | -37.055 | 90.000 | -109.235 | -129.987 |
| 31.000 | -33.082 | 90.000 | -105.202 | -129.987 |
| 32.000 | -29.566 | 90.000 | -101.630 | -129.987 |
| 33.000 | -26.798 | 90.000 | -98.814 | -129.988 |
| 34.000 | -24.685 | 90.000 | -96.659 | -129.988 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 35.000 | -23.106 | 90.000 | -95.042 | -129.988 |
| 36.000 | -21.964 | 90.000 | -93.868 | -129.988 |
| 37.000 | -21.193 | 90.000 | -93.069 | -129.989 |
| 38.000 | -20.747 | 90.000 | -92.600 | -129.989 |
| 39.000 | -20.598 | 90.000 | -92.434 | -129.989 |
| 40.000 | -20.735 | 90.000 | -92.556 | -129.989 |
| 41.000 | -21.158 | 90.000 | -92.969 | -129.989 |
| 42.000 | -21.885 | 90.000 | -93.690 | -129.989 |
| 43.000 | -22.956 | 90.000 | -94.759 | -129.989 |
| 44.000 | -24.447 | 90.000 | -96.252 | -129.989 |
| 45.000 | -26.506 | 90.000 | -98.316 | -129.990 |
| 46.000 | -29.448 | 90.000 | -101.266 | -129.990 |
| 47.000 | -34.127 | 90.000 | -105.958 | -129.990 |
| 48.000 | -44.964 | 90.000 | -116.809 | -129.990 |
| 49.000 | -42.040 | -90.000 | -113.902 | 50.009 |
| 50.000 | -33.721 | -90.000 | -105.604 | 50.009 |
| 51.000 | -29.870 | -90.000 | -101.776 | 50.009 |
| 52.000 | -27.513 | -90.000 | -99.445 | 50.009 |
| 53.000 | -25.952 | -90.000 | -97.912 | 50.009 |
| 54.000 | -24.912 | -90.000 | -96.903 | 50.009 |
| 55.000 | -24.262 | -90.000 | -96.285 | 50.009 |
| 56.000 | -23.932 | -90.000 | -95.989 | 50.009 |
| 57.000 | -23.887 | -90.000 | -95.980 | 50.009 |
| 58.000 | -24.116 | -90.000 | -96.246 | 50.009 |
| 59.000 | -24.629 | -90.000 | -96.799 | 50.008 |
| 60.000 | -25.461 | -90.000 | -97.671 | 50.008 |
| 61.000 | -26.683 | -90.000 | -98.934 | 50.008 |
| 62.000 | -28.435 | -90.000 | -100.729 | 50.008 |
| 63.000 | -31.020 | -90.000 | -103.357 | 50.008 |
| 64.000 | -35.244 | -90.000 | -107.625 | 50.008 |
| 65.000 | -45.077 | -90.000 | -117.502 | 50.008 |
| 66.000 | -43.314 | 90.000 | -115.784 | -129.992 |
| 67.000 | -34.103 | 90.000 | -106.617 | -129.992 |
| 68.000 | -29.655 | 90.000 | -102.213 | -129.993 |
| 69.000 | -26.705 | 90.000 | -99.307 | -129.992 |
| 70.000 | -24.514 | 90.000 | -97.159 | -129.993 |
| 71.000 | -22.791 | 90.000 | -95.480 | -129.993 |
| 72.000 | -21.388 | 90.000 | -94.119 | -129.993 |
| 73.000 | -20.223 | 90.000 | -92.995 | -129.993 |
| 74.000 | -19.242 | 90.000 | -92.053 | -129.993 |
| 75.000 | -18.410 | 90.000 | -91.260 | -129.993 |
| 76.000 | -17.700 | 90.000 | -90.587 | -129.993 |
| 77.000 | -17.096 | 90.000 | -90.018 | -129.993 |
| 78.000 | -16.582 | 90.000 | -89.538 | -129.993 |
| 79.000 | -16.149 | 90.000 | -89.136 | -129.993 |
| 80.000 | -15.787 | 90.000 | -88.803 | -129.993 |
| 81.000 | -15.489 | 90.000 | -88.531 | -129.993 |
| 82.000 | -15.251 | 90.000 | -88.317 | -129.994 |
| 83.000 | -15.067 | 90.000 | -88.155 | -129.994 |
| 84.000 | -14.934 | 90.000 | -88.041 | -129.994 |
| 85.000 | -14.849 | 90.000 | -87.972 | -129.994 |
| 86.000 | -14.810 | 90.000 | -87.946 | -129.994 |
| 87.000 | -14.816 | 90.000 | -87.962 | -129.994 |
| 88.000 | -14.864 | 90.000 | -88.018 | -129.994 |
| 89.000 | -14.953 | 90.000 | -88.112 | -129.994 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.514 | 90.000 | -74.179 | -89.097 |
| 1.000 | 12.476 | 90.000 | -83.262 | -84.500 |
| 2.000 | 12.359 | 90.000 | -70.625 | -10.052 |
| 3.000 | 12.165 | 90.000 | -66.949 | -169.963 |
| 4.000 | 11.892 | 90.000 | -64.777 | -169.950 |
| 5.000 | 11.539 | 90.000 | -63.315 | -169.950 |
| 6.000 | 11.104 | 90.000 | -62.314 | -169.951 |
| 7.000 | 10.586 | 90.000 | -61.576 | -169.954 |
| 8.000 | 9.982 | 90.000 | -61.175 | -169.958 |
| 9.000 | 9.290 | 90.000 | -60.946 | -169.960 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 10.000 | 8.504 | 90.000 | -60.940 | -169.962 |
| 11.000 | 7.621 | 90.000 | -61.104 | -169.965 |
| 12.000 | 6.635 | 90.000 | -61.429 | -169.967 |
| 13.000 | 5.538 | 90.000 | -61.923 | -169.969 |
| 14.000 | 4.322 | 90.000 | -62.594 | -169.970 |
| 15.000 | 2.974 | 90.000 | -63.434 | -169.971 |
| 16.000 | 1.479 | 90.000 | -64.459 | -169.972 |
| 17.000 | -0.184 | 90.000 | -65.684 | -169.974 |
| 18.000 | -2.046 | 90.000 | -67.135 | -169.975 |
| 19.000 | -4.148 | 90.000 | -68.854 | -169.975 |
| 20.000 | -6.558 | 90.000 | -70.904 | -169.976 |
| 21.000 | -9.388 | 90.000 | -73.396 | -169.977 |
| 22.000 | -12.849 | 90.000 | -76.538 | -169.978 |
| 23.000 | -17.415 | 90.000 | -80.805 | -169.979 |
| 24.000 | -24.635 | 90.000 | -87.740 | -169.980 |
| 25.000 | -55.565 | -90.000 | -118.401 | 10.020 |
| 26.000 | -26.388 | -90.000 | -88.969 | 10.020 |
| 27.000 | -21.732 | -90.000 | -84.073 | 10.019 |
| 28.000 | -19.538 | -90.000 | -81.650 | 10.019 |
| 29.000 | -18.391 | -90.000 | -80.286 | 10.018 |
| 30.000 | -17.831 | -90.000 | -79.521 | 10.018 |
| 31.000 | -17.648 | -90.000 | -79.144 | 10.018 |
| 32.000 | -17.719 | -90.000 | -79.029 | 10.017 |
| 33.000 | -17.959 | -90.000 | -79.093 | 10.017 |
| 34.000 | -18.302 | -90.000 | -79.270 | 10.017 |
| 35.000 | -18.693 | -90.000 | -79.502 | 10.016 |
| 36.000 | -19.079 | -90.000 | -79.738 | 10.016 |
| 37.000 | -19.415 | -90.000 | -79.933 | 10.015 |
| 38.000 | -19.665 | -90.000 | -80.048 | 10.015 |
| 39.000 | -19.801 | -90.000 | -80.057 | 10.015 |
| 40.000 | -19.811 | -90.000 | -79.946 | 10.015 |
| 41.000 | -19.697 | -90.000 | -79.718 | 10.015 |
| 42.000 | -19.475 | -90.000 | -79.388 | 10.015 |
| 43.000 | -19.166 | -90.000 | -78.977 | 10.014 |
| 44.000 | -18.798 | -90.000 | -78.514 | 10.014 |
| 45.000 | -18.397 | -90.000 | -78.022 | 10.014 |
| 46.000 | -17.987 | -90.000 | -77.527 | 10.014 |
| 47.000 | -17.586 | -90.000 | -77.047 | 10.014 |
| 48.000 | -17.212 | -90.000 | -76.599 | 10.013 |
| 49.000 | -16.875 | -90.000 | -76.191 | 10.013 |
| 50.000 | -16.582 | -90.000 | -75.834 | 10.013 |
| 51.000 | -16.339 | -90.000 | -75.530 | 10.013 |
| 52.000 | -16.149 | -90.000 | -75.285 | 10.012 |
| 53.000 | -16.014 | -90.000 | -75.097 | 10.012 |
| 54.000 | -15.933 | -90.000 | -74.969 | 10.012 |
| 55.000 | -15.908 | -90.000 | -74.900 | 10.012 |
| 56.000 | -15.936 | -90.000 | -74.887 | 10.012 |
| 57.000 | -16.017 | -90.000 | -74.931 | 10.012 |
| 58.000 | -16.148 | -90.000 | -75.028 | 10.012 |
| 59.000 | -16.328 | -90.000 | -75.178 | 10.011 |
| 60.000 | -16.555 | -90.000 | -75.378 | 10.011 |
| 61.000 | -16.828 | -90.000 | -75.626 | 10.011 |
| 62.000 | -17.144 | -90.000 | -75.920 | 10.011 |
| 63.000 | -17.500 | -90.000 | -76.257 | 10.011 |
| 64.000 | -17.895 | -90.000 | -76.636 | 10.011 |
| 65.000 | -18.329 | -90.000 | -77.055 | 10.011 |
| 66.000 | -18.796 | -90.000 | -77.510 | 10.011 |
| 67.000 | -19.297 | -90.000 | -78.001 | 10.011 |
| 68.000 | -19.829 | -90.000 | -78.524 | 10.010 |
| 69.000 | -20.390 | -90.000 | -79.078 | 10.011 |
| 70.000 | -20.977 | -90.000 | -79.660 | 10.010 |
| 71.000 | -21.588 | -90.000 | -80.268 | 10.010 |
| 72.000 | -22.221 | -90.000 | -80.898 | 10.010 |
| 73.000 | -22.872 | -90.000 | -81.548 | 10.010 |
| 74.000 | -23.539 | -90.000 | -82.214 | 10.010 |
| 75.000 | -24.219 | -90.000 | -82.894 | 10.010 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 76.000 | -24.907 | -90.000 | -83.584 | 10.010 |
| 77.000 | -25.601 | -90.000 | -84.278 | 10.010 |
| 78.000 | -26.293 | -90.000 | -84.973 | 10.010 |
| 79.000 | -26.982 | -90.000 | -85.664 | 10.010 |
| 80.000 | -27.659 | -90.000 | -86.343 | 10.010 |
| 81.000 | -28.319 | -90.000 | -87.006 | 10.010 |
| 82.000 | -28.955 | -90.000 | -87.644 | 10.009 |
| 83.000 | -29.559 | -90.000 | -88.251 | 10.009 |
| 84.000 | -30.125 | -90.000 | -88.819 | 10.009 |
| 85.000 | -30.643 | -90.000 | -89.340 | 10.009 |
| 86.000 | -31.108 | -90.000 | -89.806 | 10.009 |
| 87.000 | -31.511 | -90.000 | -90.211 | 10.009 |
| 88.000 | -31.847 | -90.000 | -90.548 | 10.009 |
| 89.000 | -32.112 | -90.000 | -90.813 | 10.009 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.514 | 90.000 | -76.755 | -88.651 |
| 1.000 | 12.475 | 90.000 | -90.359 | -81.755 |
| 2.000 | 12.359 | 90.000 | -71.344 | 29.917 |
| 3.000 | 12.164 | 90.000 | -65.729 | 150.050 |
| 4.000 | 11.891 | 90.000 | -62.331 | 150.070 |
| 5.000 | 11.538 | 90.000 | -59.956 | 150.071 |
| 6.000 | 11.103 | 90.000 | -58.230 | 150.069 |
| 7.000 | 10.584 | 90.000 | -56.856 | 150.063 |
| 8.000 | 9.980 | 90.000 | -55.944 | 150.058 |
| 9.000 | 9.286 | 90.000 | -55.245 | 150.055 |
| 10.000 | 8.499 | 90.000 | -54.833 | 150.051 |
| 11.000 | 7.613 | 90.000 | -54.626 | 150.048 |
| 12.000 | 6.624 | 90.000 | -54.609 | 150.045 |
| 13.000 | 5.522 | 90.000 | -54.791 | 150.042 |
| 14.000 | 4.298 | 90.000 | -55.180 | 150.040 |
| 15.000 | 2.939 | 90.000 | -55.758 | 150.038 |
| 16.000 | 1.425 | 90.000 | -56.547 | 150.037 |
| 17.000 | -0.267 | 90.000 | -57.561 | 150.035 |
| 18.000 | -2.175 | 90.000 | -58.832 | 150.034 |
| 19.000 | -4.353 | 90.000 | -60.412 | 150.032 |
| 20.000 | -6.894 | 90.000 | -62.389 | 150.031 |
| 21.000 | -9.961 | 90.000 | -64.923 | 150.030 |
| 22.000 | -13.899 | 90.000 | -68.357 | 150.029 |
| 23.000 | -19.662 | 90.000 | -73.641 | 150.028 |
| 24.000 | -32.829 | 90.000 | -86.352 | 150.027 |
| 25.000 | -27.346 | -90.000 | -80.435 | -29.974 |
| 26.000 | -20.458 | -90.000 | -73.134 | -29.975 |
| 27.000 | -17.416 | -90.000 | -69.697 | -29.975 |
| 28.000 | -15.703 | -90.000 | -67.607 | -29.976 |
| 29.000 | -14.675 | -90.000 | -66.219 | -29.977 |
| 30.000 | -14.062 | -90.000 | -65.263 | -29.977 |
| 31.000 | -13.726 | -90.000 | -64.597 | -29.977 |
| 32.000 | -13.578 | -90.000 | -64.133 | -29.978 |
| 33.000 | -13.560 | -90.000 | -63.812 | -29.979 |
| 34.000 | -13.628 | -90.000 | -63.590 | -29.979 |
| 35.000 | -13.750 | -90.000 | -63.432 | -29.979 |
| 36.000 | -13.899 | -90.000 | -63.315 | -29.980 |
| 37.000 | -14.057 | -90.000 | -63.216 | -29.980 |
| 38.000 | -14.207 | -90.000 | -63.119 | -29.981 |
| 39.000 | -14.341 | -90.000 | -63.017 | -29.981 |
| 40.000 | -14.454 | -90.000 | -62.902 | -29.982 |
| 41.000 | -14.545 | -90.000 | -62.775 | -29.982 |
| 42.000 | -14.616 | -90.000 | -62.637 | -29.982 |
| 43.000 | -14.673 | -90.000 | -62.493 | -29.982 |
| 44.000 | -14.724 | -90.000 | -62.351 | -29.982 |
| 45.000 | -14.775 | -90.000 | -62.216 | -29.983 |
| 46.000 | -14.836 | -90.000 | -62.099 | -29.983 |
| 47.000 | -14.914 | -90.000 | -62.006 | -29.983 |
| 48.000 | -15.017 | -90.000 | -61.945 | -29.984 |
| 49.000 | -15.150 | -90.000 | -61.922 | -29.984 |
| 50.000 | -15.322 | -90.000 | -61.942 | -29.984 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 51.000 | -15.536 | -90.000 | -62.012 | -29.984 |
| 52.000 | -15.797 | -90.000 | -62.136 | -29.985 |
| 53.000 | -16.111 | -90.000 | -62.317 | -29.985 |
| 54.000 | -16.479 | -90.000 | -62.560 | -29.985 |
| 55.000 | -16.907 | -90.000 | -62.867 | -29.985 |
| 56.000 | -17.399 | -90.000 | -63.244 | -29.985 |
| 57.000 | -17.959 | -90.000 | -63.693 | -29.985 |
| 58.000 | -18.591 | -90.000 | -64.221 | -29.985 |
| 59.000 | -19.303 | -90.000 | -64.833 | -29.986 |
| 60.000 | -20.101 | -90.000 | -65.536 | -29.986 |
| 61.000 | -20.995 | -90.000 | -66.340 | -29.986 |
| 62.000 | -21.998 | -90.000 | -67.257 | -29.986 |
| 63.000 | -23.127 | -90.000 | -68.305 | -29.986 |
| 64.000 | -24.406 | -90.000 | -69.507 | -29.986 |
| 65.000 | -25.871 | -90.000 | -70.899 | -29.986 |
| 66.000 | -27.576 | -90.000 | -72.535 | -29.986 |
| 67.000 | -29.609 | -90.000 | -74.503 | -29.986 |
| 68.000 | -32.129 | -90.000 | -76.962 | -29.987 |
| 69.000 | -35.467 | -90.000 | -80.242 | -29.986 |
| 70.000 | -40.517 | -90.000 | -85.237 | -29.987 |
| 71.000 | -52.100 | -90.000 | -96.771 | -29.988 |
| 72.000 | -48.064 | 90.000 | -92.687 | 150.012 |
| 73.000 | -40.109 | 90.000 | -84.688 | 150.013 |
| 74.000 | -36.314 | 90.000 | -80.853 | 150.013 |
| 75.000 | -33.898 | 90.000 | -78.399 | 150.013 |
| 76.000 | -32.184 | 90.000 | -76.650 | 150.013 |
| 77.000 | -30.901 | 90.000 | -75.334 | 150.013 |
| 78.000 | -29.908 | 90.000 | -74.313 | 150.013 |
| 79.000 | -29.131 | 90.000 | -73.508 | 150.013 |
| 80.000 | -28.515 | 90.000 | -72.869 | 150.013 |
| 81.000 | -28.030 | 90.000 | -72.363 | 150.013 |
| 82.000 | -27.652 | 90.000 | -71.965 | 150.012 |
| 83.000 | -27.364 | 90.000 | -71.661 | 150.012 |
| 84.000 | -27.154 | 90.000 | -71.436 | 150.012 |
| 85.000 | -27.012 | 90.000 | -71.282 | 150.012 |
| 86.000 | -26.932 | 90.000 | -71.192 | 150.012 |
| 87.000 | -26.908 | 90.000 | -71.160 | 150.012 |
| 88.000 | -26.936 | 90.000 | -71.184 | 150.012 |
| 89.000 | -27.015 | 90.000 | -71.259 | 150.012 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.513 | 90.000 | -79.330 | -88.204 |
| 1.000 | 12.475 | 90.000 | -97.456 | -79.010 |
| 2.000 | 12.359 | 90.000 | -72.063 | 69.886 |
| 3.000 | 12.164 | 90.000 | -64.510 | 110.063 |
| 4.000 | 11.890 | 90.000 | -59.885 | 110.090 |
| 5.000 | 11.536 | 90.000 | -56.597 | 110.091 |
| 6.000 | 11.101 | 90.000 | -54.147 | 110.088 |
| 7.000 | 10.582 | 90.000 | -52.136 | 110.081 |
| 8.000 | 9.978 | 90.000 | -50.713 | 110.075 |
| 9.000 | 9.283 | 90.000 | -49.542 | 110.070 |
| 10.000 | 8.497 | 90.000 | -48.723 | 110.065 |
| 11.000 | 7.611 | 90.000 | -48.142 | 110.061 |
| 12.000 | 6.624 | 90.000 | -47.778 | 110.057 |
| 13.000 | 5.525 | 90.000 | -47.640 | 110.053 |
| 14.000 | 4.307 | 90.000 | -47.733 | 110.051 |
| 15.000 | 2.958 | 90.000 | -48.029 | 110.048 |
| 16.000 | 1.460 | 90.000 | -48.548 | 110.046 |
| 17.000 | -0.205 | 90.000 | -49.294 | 110.044 |
| 18.000 | -2.069 | 90.000 | -50.294 | 110.042 |
| 19.000 | -4.173 | 90.000 | -51.585 | 110.040 |
| 20.000 | -6.585 | 90.000 | -53.229 | 110.038 |
| 21.000 | -9.418 | 90.000 | -55.334 | 110.037 |
| 22.000 | -12.880 | 90.000 | -58.105 | 110.036 |
| 23.000 | -17.447 | 90.000 | -62.014 | 110.035 |
| 24.000 | -24.660 | 90.000 | -68.600 | 110.033 |
| 25.000 | -56.231 | -90.000 | -99.573 | -69.967 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 26.000 | -26.457 | -90.000 | -69.227 | -69.969 |
| 27.000 | -21.796 | -90.000 | -64.018 | -69.970 |
| 28.000 | -19.603 | -90.000 | -61.300 | -69.971 |
| 29.000 | -18.457 | -90.000 | -59.651 | -69.972 |
| 30.000 | -17.900 | -90.000 | -58.612 | -69.972 |
| 31.000 | -17.720 | -90.000 | -57.967 | -69.972 |
| 32.000 | -17.794 | -90.000 | -57.593 | -69.973 |
| 33.000 | -18.037 | -90.000 | -57.407 | -69.974 |
| 34.000 | -18.384 | -90.000 | -57.339 | -69.974 |
| 35.000 | -18.778 | -90.000 | -57.333 | -69.975 |
| 36.000 | -19.167 | -90.000 | -57.338 | -69.976 |
| 37.000 | -19.507 | -90.000 | -57.307 | -69.976 |
| 38.000 | -19.760 | -90.000 | -57.201 | -69.977 |
| 39.000 | -19.899 | -90.000 | -56.996 | -69.977 |
| 40.000 | -19.913 | -90.000 | -56.675 | -69.978 |
| 41.000 | -19.803 | -90.000 | -56.243 | -69.978 |
| 42.000 | -19.584 | -90.000 | -55.713 | -69.978 |
| 43.000 | -19.280 | -90.000 | -55.108 | -69.979 |
| 44.000 | -18.917 | -90.000 | -54.454 | -69.979 |
| 45.000 | -18.520 | -90.000 | -53.776 | -69.980 |
| 46.000 | -18.115 | -90.000 | -53.100 | -69.980 |
| 47.000 | -17.719 | -90.000 | -52.442 | -69.980 |
| 48.000 | -17.350 | -90.000 | -51.820 | -69.981 |
| 49.000 | -17.017 | -90.000 | -51.243 | -69.981 |
| 50.000 | -16.730 | -90.000 | -50.719 | -69.981 |
| 51.000 | -16.491 | -90.000 | -50.254 | -69.981 |
| 52.000 | -16.307 | -90.000 | -49.849 | -69.982 |
| 53.000 | -16.177 | -90.000 | -49.507 | -69.982 |
| 54.000 | -16.101 | -90.000 | -49.227 | -69.982 |
| 55.000 | -16.081 | -90.000 | -49.009 | -69.982 |
| 56.000 | -16.113 | -90.000 | -48.852 | -69.982 |
| 57.000 | -16.198 | -90.000 | -48.754 | -69.982 |
| 58.000 | -16.334 | -90.000 | -48.713 | -69.982 |
| 59.000 | -16.519 | -90.000 | -48.729 | -69.983 |
| 60.000 | -16.750 | -90.000 | -48.798 | -69.983 |
| 61.000 | -17.027 | -90.000 | -48.919 | -69.983 |
| 62.000 | -17.347 | -90.000 | -49.088 | -69.983 |
| 63.000 | -17.707 | -90.000 | -49.305 | -69.983 |
| 64.000 | -18.106 | -90.000 | -49.567 | -69.983 |
| 65.000 | -18.543 | -90.000 | -49.872 | -69.983 |
| 66.000 | -19.015 | -90.000 | -50.218 | -69.983 |
| 67.000 | -19.519 | -90.000 | -50.603 | -69.983 |
| 68.000 | -20.054 | -90.000 | -51.024 | -69.984 |
| 69.000 | -20.618 | -90.000 | -51.479 | -69.983 |
| 70.000 | -21.208 | -90.000 | -51.966 | -69.984 |
| 71.000 | -21.822 | -90.000 | -52.484 | -69.984 |
| 72.000 | -22.458 | -90.000 | -53.027 | -69.984 |
| 73.000 | -23.111 | -90.000 | -53.594 | -69.984 |
| 74.000 | -23.780 | -90.000 | -54.183 | -69.984 |
| 75.000 | -24.462 | -90.000 | -54.788 | -69.984 |
| 76.000 | -25.152 | -90.000 | -55.408 | -69.984 |
| 77.000 | -25.847 | -90.000 | -56.036 | -69.984 |
| 78.000 | -26.541 | -90.000 | -56.671 | -69.984 |
| 79.000 | -27.231 | -90.000 | -57.304 | -69.984 |
| 80.000 | -27.908 | -90.000 | -57.932 | -69.984 |
| 81.000 | -28.569 | -90.000 | -58.547 | -69.984 |
| 82.000 | -29.206 | -90.000 | -59.142 | -69.985 |
| 83.000 | -29.811 | -90.000 | -59.712 | -69.985 |
| 84.000 | -30.376 | -90.000 | -60.246 | -69.985 |
| 85.000 | -30.895 | -90.000 | -60.739 | -69.985 |
| 86.000 | -31.359 | -90.000 | -61.181 | -69.985 |
| 87.000 | -31.763 | -90.000 | -61.568 | -69.985 |
| 88.000 | -32.099 | -90.000 | -61.892 | -69.985 |
| 89.000 | -32.363 | -90.000 | -62.149 | -69.985 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.513 | 90.000 | -79.643 | -88.204 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 1.000 | 12.475 | 90.000 | -97.769 | -79.057 |
| 2.000 | 12.359 | 90.000 | -72.376 | 109.792 |
| 3.000 | 12.163 | 90.000 | -64.822 | 109.923 |
| 4.000 | 11.889 | 90.000 | -60.198 | 109.903 |
| 5.000 | 11.535 | 90.000 | -56.909 | 109.857 |
| 6.000 | 11.100 | 90.000 | -54.459 | 109.807 |
| 7.000 | 10.580 | 90.000 | -52.447 | 109.752 |
| 8.000 | 9.976 | 90.000 | -51.024 | 109.699 |
| 9.000 | 9.281 | 90.000 | -49.852 | 109.646 |
| 10.000 | 8.496 | 90.000 | -49.030 | 109.594 |
| 11.000 | 7.613 | 90.000 | -48.446 | 109.541 |
| 12.000 | 6.629 | 90.000 | -48.077 | 109.490 |
| 13.000 | 5.537 | 90.000 | -47.930 | 109.438 |
| 14.000 | 4.332 | 90.000 | -48.009 | 109.386 |
| 15.000 | 3.003 | 90.000 | -48.283 | 109.335 |
| 16.000 | 1.538 | 90.000 | -48.767 | 109.284 |
| 17.000 | -0.074 | 90.000 | -49.458 | 109.232 |
| 18.000 | -1.851 | 90.000 | -50.368 | 109.180 |
| 19.000 | -3.813 | 90.000 | -51.516 | 109.129 |
| 20.000 | -5.989 | 90.000 | -52.920 | 109.077 |
| 21.000 | -8.414 | 90.000 | -54.615 | 109.025 |
| 22.000 | -11.135 | 90.000 | -56.643 | 108.972 |
| 23.000 | -14.216 | 90.000 | -59.063 | 108.920 |
| 24.000 | -17.745 | 90.000 | -61.962 | 108.866 |
| 25.000 | -21.838 | 90.000 | -65.453 | 108.813 |
| 26.000 | -26.619 | 90.000 | -69.659 | 108.759 |
| 27.000 | -32.083 | 90.000 | -74.573 | 108.705 |
| 28.000 | -37.386 | 90.000 | -79.347 | 108.651 |
| 29.000 | -39.553 | 90.000 | -81.007 | 108.596 |
| 30.000 | -37.106 | 90.000 | -78.074 | 108.540 |
| 31.000 | -33.178 | 90.000 | -73.678 | 108.485 |
| 32.000 | -29.690 | 90.000 | -69.738 | 108.429 |
| 33.000 | -26.941 | 90.000 | -66.555 | 108.372 |
| 34.000 | -24.843 | 90.000 | -64.038 | 108.316 |
| 35.000 | -23.276 | 90.000 | -62.068 | 108.258 |
| 36.000 | -22.146 | 90.000 | -60.548 | 108.200 |
| 37.000 | -21.386 | 90.000 | -59.413 | 108.142 |
| 38.000 | -20.952 | 90.000 | -58.615 | 108.083 |
| 39.000 | -20.814 | 90.000 | -58.128 | 108.024 |
| 40.000 | -20.963 | 90.000 | -57.937 | 107.964 |
| 41.000 | -21.397 | 90.000 | -58.044 | 107.904 |
| 42.000 | -22.137 | 90.000 | -58.467 | 107.844 |
| 43.000 | -23.221 | 90.000 | -59.245 | 107.783 |
| 44.000 | -24.729 | 90.000 | -60.456 | 107.722 |
| 45.000 | -26.807 | 90.000 | -62.247 | 107.660 |
| 46.000 | -29.777 | 90.000 | -64.941 | 107.598 |
| 47.000 | -34.511 | 90.000 | -69.407 | 107.537 |
| 48.000 | -45.627 | 90.000 | -80.264 | 107.474 |
| 49.000 | -42.077 | -90.000 | -76.463 | -72.589 |
| 50.000 | -33.927 | -90.000 | -68.070 | -72.651 |
| 51.000 | -30.121 | -90.000 | -64.031 | -72.714 |
| 52.000 | -27.789 | -90.000 | -61.472 | -72.777 |
| 53.000 | -26.246 | -90.000 | -59.710 | -72.840 |
| 54.000 | -25.220 | -90.000 | -58.474 | -72.902 |
| 55.000 | -24.582 | -90.000 | -57.631 | -72.965 |
| 56.000 | -24.261 | -90.000 | -57.113 | -73.027 |
| 57.000 | -24.224 | -90.000 | -56.886 | -73.090 |
| 58.000 | -24.460 | -90.000 | -56.939 | -73.151 |
| 59.000 | -24.979 | -90.000 | -57.281 | -73.213 |
| 60.000 | -25.813 | -90.000 | -57.947 | -73.273 |
| 61.000 | -27.034 | -90.000 | -59.004 | -73.334 |
| 62.000 | -28.779 | -90.000 | -60.592 | -73.394 |
| 63.000 | -31.342 | -90.000 | -63.004 | -73.453 |
| 64.000 | -35.501 | -90.000 | -67.020 | -73.510 |
| 65.000 | -44.958 | -90.000 | -76.336 | -73.567 |
| 66.000 | -44.274 | 90.000 | -75.521 | 106.377 |

18-31

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 67.000 | -34.742 | 90.000 | -65.862 | 106.323 |
| 68.000 | -30.232 | 90.000 | -61.231 | 106.269 |
| 69.000 | -27.261 | 90.000 | -58.144 | 106.218 |
| 70.000 | -25.060 | 90.000 | -55.834 | 106.167 |
| 71.000 | -23.333 | 90.000 | -54.004 | 106.118 |
| 72.000 | -21.931 | 90.000 | -52.502 | 106.071 |
| 73.000 | -20.766 | 90.000 | -51.246 | 106.026 |
| 74.000 | -19.787 | 90.000 | -50.180 | 105.983 |
| 75.000 | -18.958 | 90.000 | -49.268 | 105.942 |
| 76.000 | -18.251 | 90.000 | -48.486 | 105.903 |
| 77.000 | -17.650 | 90.000 | -47.814 | 105.866 |
| 78.000 | -17.139 | 90.000 | -47.238 | 105.832 |
| 79.000 | -16.709 | 90.000 | -46.747 | 105.800 |
| 80.000 | -16.349 | 90.000 | -46.333 | 105.770 |
| 81.000 | -16.055 | 90.000 | -45.989 | 105.743 |
| 82.000 | -15.819 | 90.000 | -45.708 | 105.719 |
| 83.000 | -15.637 | 90.000 | -45.488 | 105.697 |
| 84.000 | -15.506 | 90.000 | -45.323 | 105.678 |
| 85.000 | -15.423 | 90.000 | -45.212 | 105.662 |
| 86.000 | -15.385 | 90.000 | -45.150 | 105.649 |
| 87.000 | -15.392 | 90.000 | -45.139 | 105.639 |
| 88.000 | -15.440 | 90.000 | -45.174 | 105.632 |
| 89.000 | -15.531 | 90.000 | -45.256 | 105.627 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.513 | 90.000 | -77.694 | -88.649 |
| 1.000 | 12.474 | 90.000 | -91.298 | -81.895 |
| 2.000 | 12.358 | 90.000 | -72.283 | 149.637 |
| 3.000 | 12.162 | 90.000 | -66.668 | 149.629 |
| 4.000 | 11.888 | 90.000 | -63.270 | 149.508 |
| 5.000 | 11.534 | 90.000 | -60.893 | 149.368 |
| 6.000 | 11.098 | 90.000 | -59.167 | 149.224 |
| 7.000 | 10.578 | 90.000 | -57.791 | 149.077 |
| 8.000 | 9.973 | 90.000 | -56.877 | 148.930 |
| 9.000 | 9.279 | 90.000 | -56.175 | 148.783 |
| 10.000 | 8.493 | 90.000 | -55.759 | 148.637 |
| 11.000 | 7.611 | 90.000 | -55.544 | 148.490 |
| 12.000 | 6.629 | 90.000 | -55.516 | 148.343 |
| 13.000 | 5.540 | 90.000 | -55.680 | 148.195 |
| 14.000 | 4.340 | 90.000 | -56.040 | 148.047 |
| 15.000 | 3.020 | 90.000 | -56.574 | 147.899 |
| 16.000 | 1.571 | 90.000 | -57.292 | 147.750 |
| 17.000 | -0.015 | 90.000 | -58.194 | 147.600 |
| 18.000 | -1.750 | 90.000 | -59.285 | 147.449 |
| 19.000 | -3.645 | 90.000 | -60.576 | 147.298 |
| 20.000 | -5.713 | 90.000 | -62.072 | 147.146 |
| 21.000 | -7.961 | 90.000 | -63.779 | 146.993 |
| 22.000 | -10.388 | 90.000 | -65.694 | 146.838 |
| 23.000 | -12.973 | 90.000 | -67.791 | 146.683 |
| 24.000 | -15.652 | 90.000 | -70.005 | 146.526 |
| 25.000 | -18.275 | 90.000 | -72.185 | 146.368 |
| 26.000 | -20.572 | 90.000 | -74.059 | 146.210 |
| 27.000 | -22.171 | 90.000 | -75.255 | 146.050 |
| 28.000 | -22.785 | 90.000 | -75.481 | 145.888 |
| 29.000 | -22.434 | 90.000 | -74.760 | 145.725 |
| 30.000 | -21.436 | 90.000 | -73.407 | 145.561 |
| 31.000 | -20.157 | 90.000 | -71.786 | 145.396 |
| 32.000 | -18.851 | 90.000 | -70.152 | 145.229 |
| 33.000 | -17.658 | 90.000 | -68.643 | 145.061 |
| 34.000 | -16.635 | 90.000 | -67.317 | 144.891 |
| 35.000 | -15.803 | 90.000 | -66.193 | 144.720 |
| 36.000 | -15.161 | 90.000 | -65.271 | 144.547 |
| 37.000 | -14.706 | 90.000 | -64.544 | 144.374 |
| 38.000 | -14.425 | 90.000 | -64.002 | 144.198 |
| 39.000 | -14.310 | 90.000 | -63.637 | 144.022 |
| 40.000 | -14.353 | 90.000 | -63.437 | 143.844 |
| 41.000 | -14.547 | 90.000 | -63.397 | 143.665 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 42.000 | -14.886 | 90.000 | -63.510 | 143.485 |
| 43.000 | -15.364 | 90.000 | -63.771 | 143.303 |
| 44.000 | -15.978 | 90.000 | -64.174 | 143.120 |
| 45.000 | -16.721 | 90.000 | -64.715 | 142.936 |
| 46.000 | -17.587 | 90.000 | -65.386 | 142.751 |
| 47.000 | -18.567 | 90.000 | -66.177 | 142.566 |
| 48.000 | -19.646 | 90.000 | -67.073 | 142.379 |
| 49.000 | -20.796 | 90.000 | -68.048 | 142.192 |
| 50.000 | -21.979 | 90.000 | -69.061 | 142.005 |
| 51.000 | -23.133 | 90.000 | -70.052 | 141.817 |
| 52.000 | -24.174 | 90.000 | -70.936 | 141.629 |
| 53.000 | -24.996 | 90.000 | -71.606 | 141.441 |
| 54.000 | -25.491 | 90.000 | -71.955 | 141.253 |
| 55.000 | -25.580 | 90.000 | -71.902 | 141.066 |
| 56.000 | -25.244 | 90.000 | -71.430 | 140.879 |
| 57.000 | -24.533 | 90.000 | -70.588 | 140.692 |
| 58.000 | -23.542 | 90.000 | -69.471 | 140.508 |
| 59.000 | -22.378 | 90.000 | -68.186 | 140.324 |
| 60.000 | -21.132 | 90.000 | -66.824 | 140.142 |
| 61.000 | -19.871 | 90.000 | -65.451 | 139.961 |
| 62.000 | -18.640 | 90.000 | -64.113 | 139.782 |
| 63.000 | -17.464 | 90.000 | -62.835 | 139.606 |
| 64.000 | -16.359 | 90.000 | -61.631 | 139.433 |
| 65.000 | -15.331 | 90.000 | -60.508 | 139.263 |
| 66.000 | -14.381 | 90.000 | -59.469 | 139.095 |
| 67.000 | -13.509 | 90.000 | -58.510 | 138.932 |
| 68.000 | -12.711 | 90.000 | -57.630 | 138.772 |
| 69.000 | -11.983 | 90.000 | -56.825 | 138.617 |
| 70.000 | -11.322 | 90.000 | -56.090 | 138.466 |
| 71.000 | -10.725 | 90.000 | -55.423 | 138.319 |
| 72.000 | -10.187 | 90.000 | -54.818 | 138.179 |
| 73.000 | -9.703 | 90.000 | -54.273 | 138.044 |
| 74.000 | -9.272 | 90.000 | -53.783 | 137.914 |
| 75.000 | -8.891 | 90.000 | -53.346 | 137.791 |
| 76.000 | -8.554 | 90.000 | -52.959 | 137.674 |
| 77.000 | -8.262 | 90.000 | -52.619 | 137.564 |
| 78.000 | -8.011 | 90.000 | -52.324 | 137.461 |
| 79.000 | -7.800 | 90.000 | -52.072 | 137.365 |
| 80.000 | -7.625 | 90.000 | -51.861 | 137.276 |
| 81.000 | -7.487 | 90.000 | -51.689 | 137.195 |
| 82.000 | -7.382 | 90.000 | -51.554 | 137.122 |
| 83.000 | -7.310 | 90.000 | -51.457 | 137.058 |
| 84.000 | -7.270 | 90.000 | -51.394 | 137.001 |
| 85.000 | -7.262 | 90.000 | -51.366 | 136.953 |
| 86.000 | -7.283 | 90.000 | -51.371 | 136.914 |
| 87.000 | -7.334 | 90.000 | -51.410 | 136.883 |
| 88.000 | -7.414 | 90.000 | -51.481 | 136.861 |
| 89.000 | -7.523 | 90.000 | -51.585 | 136.847 |
| 90.000 | -292.626 | 0.000 | -292.626 | 0.000 |
| 0.000 | 12.512 | 90.000 | -75.745 | -89.095 |
| 1.000 | 12.474 | 90.000 | -84.827 | -84.732 |
| 2.000 | 12.357 | 90.000 | -72.190 | -170.519 |
| 3.000 | 12.162 | 90.000 | -68.514 | -170.665 |
| 4.000 | 11.887 | 90.000 | -66.341 | -170.887 |
| 5.000 | 11.533 | 90.000 | -64.878 | -171.121 |
| 6.000 | 11.096 | 90.000 | -63.875 | -171.359 |
| 7.000 | 10.576 | 90.000 | -63.135 | -171.598 |
| 8.000 | 9.970 | 90.000 | -62.732 | -171.839 |
| 9.000 | 9.274 | 90.000 | -62.500 | -172.079 |
| 10.000 | 8.487 | 90.000 | -62.490 | -172.320 |
| 11.000 | 7.603 | 90.000 | -62.649 | -172.561 |
| 12.000 | 6.618 | 90.000 | -62.966 | -172.804 |
| 13.000 | 5.524 | 90.000 | -63.449 | -173.048 |
| 14.000 | 4.316 | 90.000 | -64.103 | -173.292 |
| 15.000 | 2.985 | 90.000 | -64.917 | -173.537 |
| 16.000 | 1.518 | 90.000 | -65.904 | -173.783 |

18-33

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 17.000 | -0.096 | 90.000 | -67.070 | -174.032 |
| 18.000 | -1.875 | 90.000 | -68.428 | -174.282 |
| 19.000 | -3.840 | 90.000 | -69.998 | -174.533 |
| 20.000 | -6.018 | 90.000 | -71.804 | -174.785 |
| 21.000 | -8.446 | 90.000 | -73.881 | -175.040 |
| 22.000 | -11.169 | 90.000 | -76.272 | -175.296 |
| 23.000 | -14.252 | 90.000 | -79.040 | -175.554 |
| 24.000 | -17.783 | 90.000 | -82.272 | -175.814 |
| 25.000 | -21.875 | 90.000 | -86.080 | -176.076 |
| 26.000 | -26.652 | 90.000 | -90.587 | -176.340 |
| 27.000 | -32.102 | 90.000 | -95.779 | -176.606 |
| 28.000 | -37.373 | 90.000 | -100.804 | -176.875 |
| 29.000 | -39.517 | 90.000 | -102.713 | -177.145 |
| 30.000 | -37.091 | 90.000 | -100.065 | -177.418 |
| 31.000 | -33.192 | 90.000 | -95.951 | -177.693 |
| 32.000 | -29.723 | 90.000 | -92.277 | -177.970 |
| 33.000 | -26.987 | 90.000 | -89.344 | -178.251 |
| 34.000 | -24.899 | 90.000 | -87.068 | -178.533 |
| 35.000 | -23.339 | 90.000 | -85.328 | -178.818 |
| 36.000 | -22.215 | 90.000 | -84.032 | -179.105 |
| 37.000 | -21.461 | 90.000 | -83.112 | -179.394 |
| 38.000 | -21.031 | 90.000 | -82.523 | -179.686 |
| 39.000 | -20.898 | 90.000 | -82.238 | -179.979 |
| 40.000 | -21.050 | 90.000 | -82.243 | 179.724 |
| 41.000 | -21.488 | 90.000 | -82.541 | 179.426 |
| 42.000 | -22.229 | 90.000 | -83.148 | 179.125 |
| 43.000 | -23.314 | 90.000 | -84.104 | 178.822 |
| 44.000 | -24.819 | 90.000 | -85.485 | 178.519 |
| 45.000 | -26.891 | 90.000 | -87.438 | 178.212 |
| 46.000 | -29.845 | 90.000 | -90.278 | 177.904 |
| 47.000 | -34.531 | 90.000 | -94.855 | 177.595 |
| 48.000 | -45.337 | 90.000 | -105.555 | 177.285 |
| 49.000 | -42.538 | -90.000 | -102.656 | -3.027 |
| 50.000 | -34.205 | -90.000 | -94.226 | -3.339 |
| 51.000 | -30.362 | -90.000 | -90.291 | -3.652 |
| 52.000 | -28.017 | -90.000 | -87.856 | -3.965 |
| 53.000 | -26.468 | -90.000 | -86.222 | -4.278 |
| 54.000 | -25.441 | -90.000 | -85.114 | -4.591 |
| 55.000 | -24.804 | -90.000 | -84.399 | -4.903 |
| 56.000 | -24.486 | -90.000 | -84.006 | -5.214 |
| 57.000 | -24.453 | -90.000 | -83.900 | -5.525 |
| 58.000 | -24.693 | -90.000 | -84.072 | -5.833 |
| 59.000 | -25.217 | -90.000 | -84.530 | -6.140 |
| 60.000 | -26.058 | -90.000 | -85.308 | -6.443 |
| 61.000 | -27.286 | -90.000 | -86.477 | -6.744 |
| 62.000 | -29.041 | -90.000 | -88.175 | -7.042 |
| 63.000 | -31.620 | -90.000 | -90.698 | -7.335 |
| 64.000 | -35.811 | -90.000 | -94.838 | -7.624 |
| 65.000 | -45.415 | -90.000 | -104.390 | -7.908 |
| 66.000 | -44.327 | 90.000 | -103.256 | 171.814 |
| 67.000 | -34.927 | 90.000 | -93.811 | 171.541 |
| 68.000 | -30.448 | 90.000 | -89.289 | 171.275 |
| 69.000 | -27.491 | 90.000 | -86.291 | 171.016 |
| 70.000 | -25.300 | 90.000 | -84.062 | 170.765 |
| 71.000 | -23.581 | 90.000 | -82.306 | 170.521 |
| 72.000 | -22.184 | 90.000 | -80.875 | 170.286 |
| 73.000 | -21.025 | 90.000 | -79.684 | 170.061 |
| 74.000 | -20.051 | 90.000 | -78.680 | 169.845 |
| 75.000 | -19.225 | 90.000 | -77.825 | 169.640 |
| 76.000 | -18.522 | 90.000 | -77.097 | 169.445 |
| 77.000 | -17.925 | 90.000 | -76.474 | 169.261 |
| 78.000 | -17.417 | 90.000 | -75.944 | 169.090 |
| 79.000 | -16.989 | 90.000 | -75.495 | 168.930 |
| 80.000 | -16.632 | 90.000 | -75.120 | 168.782 |
| 81.000 | -16.340 | 90.000 | -74.810 | 168.648 |
| 82.000 | -16.106 | 90.000 | -74.561 | 168.526 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 83.000 | -15.926 | 90.000 | -74.367 | 168.418 |
| 84.000 | -15.796 | 90.000 | -74.226 | 168.324 |
| 85.000 | -15.714 | 90.000 | -74.135 | 168.244 |
| 86.000 | -15.678 | 90.000 | -74.090 | 168.178 |
| 87.000 | -15.685 | 90.000 | -74.091 | 168.126 |
| 88.000 | -15.734 | 90.000 | -74.136 | 168.090 |
| 89.000 | -15.825 | 90.000 | -74.224 | 168.068 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.512 | 90.000 | -73.796 | -89.540 |
| 1.000 | 12.473 | 90.000 | -78.357 | -87.570 |
| 2.000 | 12.357 | 90.000 | -72.098 | -130.674 |
| 3.000 | 12.161 | 90.000 | -70.359 | -130.959 |
| 4.000 | 11.886 | 90.000 | -69.413 | -131.281 |
| 5.000 | 11.531 | 90.000 | -68.862 | -131.610 |
| 6.000 | 11.094 | 90.000 | -68.583 | -131.941 |
| 7.000 | 10.573 | 90.000 | -68.478 | -132.274 |
| 8.000 | 9.966 | 90.000 | -68.586 | -132.607 |
| 9.000 | 9.269 | 90.000 | -68.826 | -132.942 |
| 10.000 | 8.479 | 90.000 | -69.224 | -133.277 |
| 11.000 | 7.592 | 90.000 | -69.756 | -133.613 |
| 12.000 | 6.601 | 90.000 | -70.421 | -133.951 |
| 13.000 | 5.498 | 90.000 | -71.227 | -134.290 |
| 14.000 | 4.276 | 90.000 | -72.182 | -134.631 |
| 15.000 | 2.922 | 90.000 | -73.288 | -134.973 |
| 16.000 | 1.421 | 90.000 | -74.560 | -135.317 |
| 17.000 | -0.250 | 90.000 | -76.018 | -135.664 |
| 18.000 | -2.118 | 90.000 | -77.688 | -136.013 |
| 19.000 | -4.227 | 90.000 | -79.613 | -136.363 |
| 20.000 | -6.644 | 90.000 | -81.857 | -136.716 |
| 21.000 | -9.481 | 90.000 | -84.533 | -137.072 |
| 22.000 | -12.947 | 90.000 | -87.849 | -137.430 |
| 23.000 | -17.517 | 90.000 | -92.275 | -137.790 |
| 24.000 | -24.723 | 90.000 | -99.348 | -138.154 |
| 25.000 | -57.356 | -90.001 | -131.856 | 41.483 |
| 26.000 | -26.592 | -90.000 | -100.973 | 41.111 |
| 27.000 | -21.925 | -90.000 | -96.196 | 40.739 |
| 28.000 | -19.736 | -90.000 | -93.902 | 40.363 |
| 29.000 | -18.597 | -90.000 | -92.665 | 39.984 |
| 30.000 | -18.049 | -90.000 | -92.024 | 39.602 |
| 31.000 | -17.878 | -90.000 | -91.767 | 39.218 |
| 32.000 | -17.962 | -90.000 | -91.767 | 38.830 |
| 33.000 | -18.216 | -90.000 | -91.944 | 38.438 |
| 34.000 | -18.574 | -90.000 | -92.229 | 38.042 |
| 35.000 | -18.979 | -90.000 | -92.566 | 37.644 |
| 36.000 | -19.381 | -90.000 | -92.904 | 37.242 |
| 37.000 | -19.733 | -90.000 | -93.196 | 36.838 |
| 38.000 | -19.998 | -90.000 | -93.404 | 36.429 |
| 39.000 | -20.150 | -90.000 | -93.502 | 36.019 |
| 40.000 | -20.174 | -90.000 | -93.478 | 35.604 |
| 41.000 | -20.076 | -90.000 | -93.332 | 35.186 |
| 42.000 | -19.867 | -90.000 | -93.080 | 34.766 |
| 43.000 | -19.572 | -90.000 | -92.745 | 34.342 |
| 44.000 | -19.218 | -90.000 | -92.353 | 33.917 |
| 45.000 | -18.830 | -90.000 | -91.931 | 33.488 |
| 46.000 | -18.433 | -90.000 | -91.501 | 33.057 |
| 47.000 | -18.047 | -90.000 | -91.085 | 32.625 |
| 48.000 | -17.687 | -90.000 | -90.696 | 32.190 |
| 49.000 | -17.363 | -90.000 | -90.347 | 31.754 |
| 50.000 | -17.084 | -90.000 | -90.044 | 31.317 |
| 51.000 | -16.855 | -90.000 | -89.793 | 30.879 |
| 52.000 | -16.680 | -90.000 | -89.597 | 30.441 |
| 53.000 | -16.558 | -90.000 | -89.458 | 30.002 |
| 54.000 | -16.492 | -90.000 | -89.375 | 29.564 |
| 55.000 | -16.481 | -90.000 | -89.349 | 29.127 |
| 56.000 | -16.523 | -90.000 | -89.377 | 28.692 |
| 57.000 | -16.618 | -90.000 | -89.458 | 28.257 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 58.000 | -16.763 | -90.000 | -89.592 | 27.826 |
| 59.000 | -16.956 | -90.000 | -89.775 | 27.397 |
| 60.000 | -17.197 | -90.000 | -90.007 | 26.972 |
| 61.000 | -17.483 | -90.000 | -90.284 | 26.550 |
| 62.000 | -17.812 | -90.000 | -90.605 | 26.134 |
| 63.000 | -18.182 | -90.000 | -90.968 | 25.723 |
| 64.000 | -18.590 | -90.000 | -91.371 | 25.319 |
| 65.000 | -19.036 | -90.000 | -91.810 | 24.922 |
| 66.000 | -19.516 | -90.000 | -92.286 | 24.532 |
| 67.000 | -20.029 | -90.000 | -92.794 | 24.151 |
| 68.000 | -20.572 | -90.000 | -93.334 | 23.778 |
| 69.000 | -21.144 | -90.000 | -93.902 | 23.416 |
| 70.000 | -21.742 | -90.000 | -94.498 | 23.063 |
| 71.000 | -22.364 | -90.000 | -95.116 | 22.722 |
| 72.000 | -23.007 | -90.000 | -95.757 | 22.394 |
| 73.000 | -23.668 | -90.000 | -96.416 | 22.079 |
| 74.000 | -24.344 | -90.000 | -97.091 | 21.777 |
| 75.000 | -25.033 | -90.000 | -97.777 | 21.489 |
| 76.000 | -25.729 | -90.000 | -98.473 | 21.217 |
| 77.000 | -26.431 | -90.000 | -99.173 | 20.959 |
| 78.000 | -27.131 | -90.000 | -99.872 | 20.719 |
| 79.000 | -27.825 | -90.000 | -100.565 | 20.495 |
| 80.000 | -28.509 | -90.000 | -101.248 | 20.288 |
| 81.000 | -29.175 | -90.000 | -101.914 | 20.100 |
| 82.000 | -29.816 | -90.000 | -102.554 | 19.929 |
| 83.000 | -30.425 | -90.000 | -103.162 | 19.779 |
| 84.000 | -30.995 | -90.000 | -103.731 | 19.646 |
| 85.000 | -31.517 | -90.000 | -104.253 | 19.534 |
| 86.000 | -31.984 | -90.000 | -104.720 | 19.443 |
| 87.000 | -32.390 | -90.000 | -105.125 | 19.370 |
| 88.000 | -32.728 | -90.000 | -105.463 | 19.319 |
| 89.000 | -32.993 | -90.000 | -105.728 | 19.288 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.511 | 90.000 | -300.000 | 0.000 |
| 1.000 | 12.473 | 90.000 | -300.000 | 0.000 |
| 2.000 | 12.356 | 90.000 | -300.000 | 0.000 |
| 3.000 | 12.160 | 90.000 | -300.000 | 0.000 |
| 4.000 | 11.885 | 90.000 | -300.000 | 0.000 |
| 5.000 | 11.530 | 90.000 | -300.000 | 0.000 |
| 6.000 | 11.092 | 90.000 | -300.000 | 0.000 |
| 7.000 | 10.571 | 90.000 | -300.000 | 0.000 |
| 8.000 | 9.962 | 90.000 | -300.000 | 0.000 |
| 9.000 | 9.265 | 90.000 | -300.000 | 0.000 |
| 10.000 | 8.473 | 90.000 | -300.000 | 0.000 |
| 11.000 | 7.584 | 90.000 | -300.000 | 0.000 |
| 12.000 | 6.590 | 90.000 | -300.000 | 0.000 |
| 13.000 | 5.482 | 90.000 | -300.000 | 0.000 |
| 14.000 | 4.252 | 90.000 | -300.000 | 0.000 |
| 15.000 | 2.887 | 90.000 | -300.000 | 0.000 |
| 16.000 | 1.367 | 90.000 | -300.000 | 0.000 |
| 17.000 | -0.333 | 90.000 | -300.000 | 0.000 |
| 18.000 | -2.247 | 90.000 | -300.000 | 0.000 |
| 19.000 | -4.433 | 90.000 | -300.000 | 0.000 |
| 20.000 | -6.982 | 90.000 | -300.000 | 0.000 |
| 21.000 | -10.056 | 90.000 | -300.000 | 0.000 |
| 22.000 | -14.000 | 90.000 | -300.000 | 0.000 |
| 23.000 | -19.763 | 90.000 | -300.000 | 0.000 |
| 24.000 | -32.870 | 90.000 | -300.000 | 0.000 |
| 25.000 | -27.536 | -90.000 | -300.000 | 0.000 |
| 26.000 | -20.633 | -90.000 | -300.000 | 0.000 |
| 27.000 | -17.594 | -90.000 | -300.000 | 0.000 |
| 28.000 | -15.891 | -90.000 | -300.000 | 0.000 |
| 29.000 | -14.873 | -90.000 | -300.000 | 0.000 |
| 30.000 | -14.274 | -90.000 | -300.000 | 0.000 |
| 31.000 | -13.950 | -90.000 | -300.000 | 0.000 |
| 32.000 | -13.815 | -90.000 | -300.000 | 0.000 |

TABLE 5 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 33.000 | -13.810 | -90.000 | -300.000 | 0.000 |
| 34.000 | -13.893 | -90.000 | -300.000 | 0.000 |
| 35.000 | -14.029 | -90.000 | -300.000 | 0.000 |
| 36.000 | -14.193 | -90.000 | -300.000 | 0.000 |
| 37.000 | -14.367 | -90.000 | -300.000 | 0.000 |
| 38.000 | -14.532 | -90.000 | -300.000 | 0.000 |
| 39.000 | -14.682 | -90.000 | -300.000 | 0.000 |
| 40.000 | -14.810 | -90.000 | -300.000 | 0.000 |
| 41.000 | -14.917 | -90.000 | -300.000 | 0.000 |
| 42.000 | -15.003 | -90.000 | -300.000 | 0.000 |
| 43.000 | -15.076 | -90.000 | -300.000 | 0.000 |
| 44.000 | -15.143 | -90.000 | -300.000 | 0.000 |
| 45.000 | -15.210 | -90.000 | -300.000 | 0.000 |
| 46.000 | -15.286 | -90.000 | -300.000 | 0.000 |
| 47.000 | -15.380 | -90.000 | -300.000 | 0.000 |
| 48.000 | -15.498 | -90.000 | -300.000 | 0.000 |
| 49.000 | -15.647 | -90.000 | -300.000 | 0.000 |
| 50.000 | -15.835 | -90.000 | -300.000 | 0.000 |
| 51.000 | -16.064 | -90.000 | -300.000 | 0.000 |
| 52.000 | -16.342 | -90.000 | -300.000 | 0.000 |
| 53.000 | -16.670 | -90.000 | -300.000 | 0.000 |
| 54.000 | -17.055 | -90.000 | -300.000 | 0.000 |
| 55.000 | -17.499 | -90.000 | -300.000 | 0.000 |
| 56.000 | -18.007 | -90.000 | -300.000 | 0.000 |
| 57.000 | -18.582 | -90.000 | -300.000 | 0.000 |
| 58.000 | -19.230 | -90.000 | -300.000 | 0.000 |
| 59.000 | -19.958 | -90.000 | -300.000 | 0.000 |
| 60.000 | -20.772 | -90.000 | -300.000 | 0.000 |
| 61.000 | -21.682 | -90.000 | -300.000 | 0.000 |
| 62.000 | -22.702 | -90.000 | -300.000 | 0.000 |
| 63.000 | -23.848 | -90.000 | -300.000 | 0.000 |
| 64.000 | -25.146 | -90.000 | -300.000 | 0.000 |
| 65.000 | -26.631 | -90.000 | -300.000 | 0.000 |
| 66.000 | -28.357 | -90.000 | -300.000 | 0.000 |
| 67.000 | -30.416 | -90.000 | -300.000 | 0.000 |
| 68.000 | -32.969 | -90.000 | -300.000 | 0.000 |
| 69.000 | -36.357 | -90.000 | -300.000 | 0.000 |
| 70.000 | -41.515 | -90.000 | -300.000 | 0.000 |
| 71.000 | -53.753 | -90.000 | -300.000 | 0.000 |
| 72.000 | -48.364 | 90.000 | -300.000 | 0.000 |
| 73.000 | -40.717 | 90.000 | -300.000 | 0.000 |
| 74.000 | -37.003 | 90.000 | -300.000 | 0.000 |
| 75.000 | -34.629 | 90.000 | -300.000 | 0.000 |
| 76.000 | -32.941 | 90.000 | -300.000 | 0.000 |
| 77.000 | -31.677 | 90.000 | -300.000 | 0.000 |
| 78.000 | -30.701 | 90.000 | -300.000 | 0.000 |
| 79.000 | -29.934 | 90.000 | -300.000 | 0.000 |
| 80.000 | -29.330 | 90.000 | -300.000 | 0.000 |
| 81.000 | -28.854 | 90.000 | -300.000 | 0.000 |
| 82.000 | -28.484 | 90.000 | -300.000 | 0.000 |
| 83.000 | -28.202 | 90.000 | -300.000 | 0.000 |
| 84.000 | -27.998 | 90.000 | -300.000 | 0.000 |
| 85.000 | -27.861 | 90.000 | -300.000 | 0.000 |
| 86.000 | -27.784 | 90.000 | -300.000 | 0.000 |
| 87.000 | -27.763 | 90.000 | -300.000 | 0.000 |
| 88.000 | -27.794 | 90.000 | -300.000 | 0.000 |
| 89.000 | -27.874 | 90.000 | -300.000 | 0.000 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |

2) Step 2: The reflector pattern is calculated by using the array feed as a single composite feed. The input data are given in Table 6.

# TABLE 6

## INPUT DATA FOR CALCULATING THE QUADRANT REFLECTOR PATTERNS
## (METHOD B)

```
CM:       ***** 19FPX.DAT *****
CM:       --- TILT= 21.6 DEG. ---
CM:          --- AIC ONLY ---
CM: DEFOCUSSED 19 ELEMENTS ARRAY FEED PATTERN
CE:   ARRAY PATTERN OBTAINED FROM RF19F2.DAT
DG:    EDGES ROTATED 90 DEGREES IN AF:
1
3   137.7      3.   3.   0      22
 -70.22      78.02
 -74.58      74.58
 -80.79      67.79
 -86.39      60.49
 -91.34      52.73
 -95.58      44.57
 -99.11      36.07
-101.87      27.30
-103.86      18.31
-105.06       9.19
-105.47       0.00
-105.06      -9.19
-103.86     -18.31
-101.87     -27.30
 -99.11     -36.07
 -95.58     -44.57
 -91.34     -52.73
 -86.39     -60.49
 -80.79     -67.79
 -74.58     -74.58
 -70.22     -78.02
   7.80       0.00
TO:
F    0.    0.
F    0     0     0     0
F    F     F     F     0
T    T     F    0.8
T    F     F     F     F     F     0.    0.
F    F    0.    0.
FD:
0    T
T    0    T    1    90.    0.    F
10   0. 10. 20. 30. 40. 50. 60. 70. 80. 90.
91
          0.000       12.514       90.000      -300.000       0.000
          1.000       12.476       90.000      -300.000       0.000
          2.000       12.360       90.000      -300.000       0.000
          3.000       12.166       90.000      -300.000       0.000
          4.000       11.894       90.000      -300.000       0.000
            .            .            .            .            .
            .            .            .            .            .
            .            .            .            .            .

                        ( see Table 5. )

            .            .            .            .            .
            .            .            .            .            .
         86.000      -27.784       90.000      -300.000       0.000
         87.000      -27.763       90.000      -300.000       0.000
         88.000      -27.794       90.000      -300.000       0.000
         89.000      -27.874       90.000      -300.000       0.000
```

TABLE 6 - CONTINUED

```
        90.000    -300.000      0.000    -300.000      0.000
AF: DEFOCUSED SINGLE COMPOSITE FEED
1
7.8    7.8    137.7
0.    0.    0.
1.    0.
90.  21.6  90.
FQ:
1    15.
PZ:
1
-135.
-12.    12.    0.05
F
PP:
1 .
1   1
1   2
XQ:
```

The resulting reflector pattern with the correct gain level for the $\phi$ = -135° cut is shown in Figure 7. Note that a good agreement is achieved between the results from the two methods because the reflector is in the far field region of the array feed. It takes 297.9 seconds cpu time for method A and 73.4 seconds cpu time for method B for a pattern with 480 pattern points and grid size $D_x$ = $D_y$ = 3 inches.
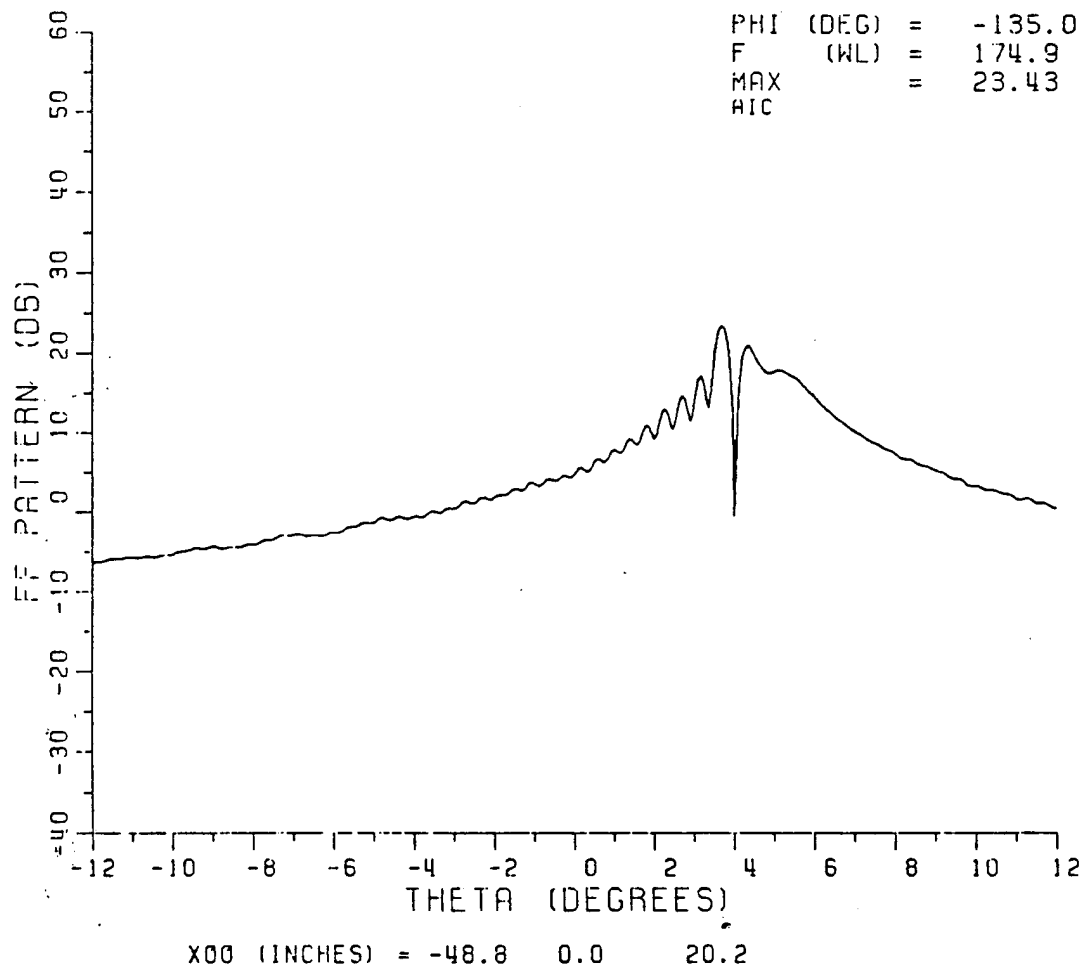


Figure 7. The reflector pattern for the $\phi$ = -135° cut by using method B.

18-41

---------------------------------------------------------
Command TO:   BASIC TEST OPTIONS

---------------------------------------------------------


This command enables the user to obtain an extended output of various intermediate quantities in the computer code.  This is useful in testing the program or in analyzing the contributions from various scattering mechanisms in terms of the total solution.


BLOCK DIAGRAM FOR TEST OPTIONS


|Command TO:|
|
|Read:  LWFD,APSIF,ADPSI|
|
|READ:  LWEA,MEAI,MEAF,NEAI,NEAF|
|
|Read:  LDEBUG,LTEST,LWYSUM,LOUT,NTEST2|
|
|Read:  LSLOPE,LCORNR,LDIP,BDEL|
|
|Read:  LAIC,LAIS,LPO,LFEED,LGTD,LREFL,THETAX,ZXP|
|
|Read:  LMP,LSR,THEXF,THEXR|


19-1
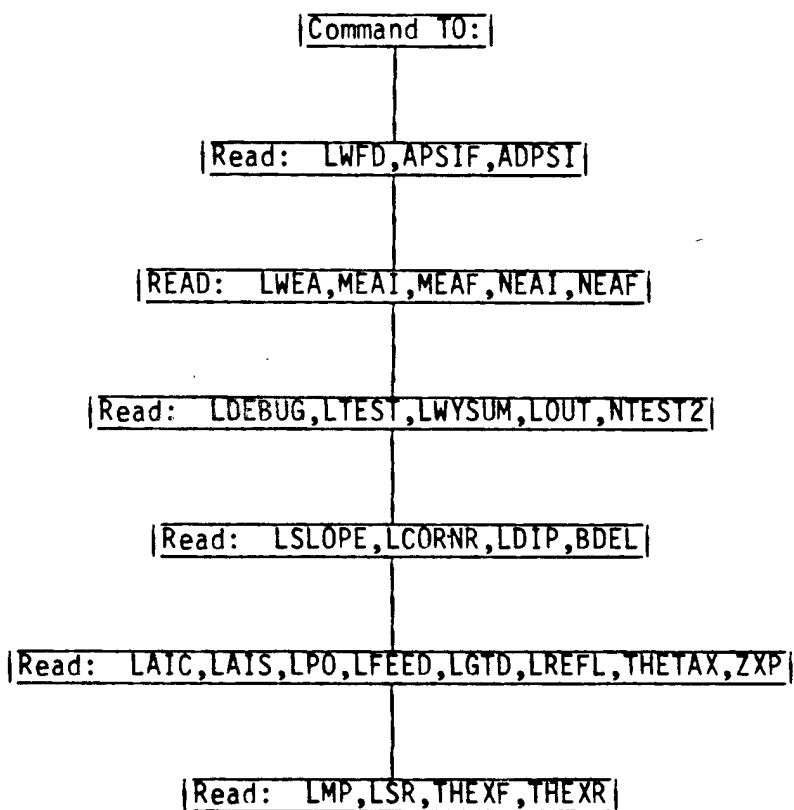
1. READ: LWFD,APSIF,ADPSI

   This read statement is used to output feed pattern data.

   a) LWFD:  This logical variable is used to tell the code whether or not to calculate and output data for the feed pattern.  (normally set true)

   b) APSIF:  This real variable defines the final feed pattern angle (in degrees) for output.

   c) ADPSI:  This real variable defines the angle (in degrees) by which output data for the feed pattern is to be incremented.


2. READ: LWEA,MEAI,MEAF,NEAI,NEAF

   This read statement is used to output data for the aperture field

   a) LWEA:  This logical variable (normally set false) is used to tell the code whether or not to output aperture field data in any rectangular subarray as follows:

   b) MEAI,MEAF:  These real variables define the initial and final values of the vertical grid lines to output data for the aperture field.

   c) NEAI,NEAF:  These real variables define the initial and final values of the horizontal grid lines to output data for the aperture field.


3. READ: LDEBUG,LTEST,LWYSUM,LOUT,NTEST2

   a) LDEBUG:  This logical variable is used to debug the program if errors are suspected within the program.  If set true, the program prints out data on unit #6 associated with each of its internal operations. These data can, then, be compared with previous data which are known to be correct. It is also used to insure initial operation of the code. Only one pattern angle is usually considered.  (normally set false)

b) LTEST: This logical variable is used to test key variables in the code such as the input/output associated with each subroutine. The data written out on Unit #6 are associated with the data in the window of the subroutine. They are written out each time the subroutine is called. It is, also, used to insure initial operation of the code. Only one pattern angle is considered. (normally set false)

c) LWYSUM: This logical variable is used to output data about the aperture field and the partial sums of the aperture integration including the y-integration YSUM data. This data is controlled separately from that controlled by LDEBUG or LTEST because of the large amount of output. (normally set false)

d) LOUT: This logical variable is used to output data on unit #6 associated with the main program. It is also used to initially insure proper operation. It can be used to examine the various components of the pattern. (normally set false)

e) NTEST2: This integer variable is used to activate certain write statements for detailed checks or debugging. The value of NTEST2 controls the detailed level to which write statements are activated. Presently, NTEST2=0 unless this command is used.

4. READ: LSLOPE, LCORNR,LDIP,BDEL

These logical variables allow certain GTD diffraction terms to be suppressed for test purposes.

a) LSLOPE: This logical variable is used to tell the code whether or not slope diffraction is desired during the computation. (normally set true)

b) LCORNR: This logical variable is used to tell the code whether or not corner diffraction is desired during the computation. (normally set to true)

c) LDIP: This logical variable is used to tell the code whether or not only the part (DI+) of the GTD diffraction coefficient, which corresponds to the aperture field, is desired during the computation. (normally set false)

d) BDEL: This real variable is used in the subroutines to adjust the bounds for corner diffraction from the reflector rim. Normally, $0.5 < BDEL < 0.8$; smaller values in this range may improve runtimes, at some loss of accuracy. Presently BDEL=0.8, unless this command is used.

5. READ: LAIC,LAIS,LPO,LFEED,LGTD,LREFL,THETAX,ZXP

  a) LAIC: This logical variable is used to tell the code whether or not conventional aperture integration (AIC) is to be used in computing the pattern. (Normally set true) When NTYPE=10 or 11, LAIC must be false.

  b) LAIS: This logical variable is used to tell the code whether or not the aperture integration on the reflector surface (AIS) is to be used in computing the patterns. (Normally set false)

  c) LPO: This logical variable is used to tell the code whether or not the physical optics (PO) or surface current method is to be used in computing the patterns. (Normally set false)

  d) LFEED: This logical variable is used to tell the code whether or not the primary feed pattern spillover is desired during the pattern computation. (normally set true)

  e) LGTD: This logical variable is used to tell the code whether or not GTD is to be used in computing the pattern. If set false only aperture integration can be used for the contribution from the reflector. (normally set true)

  f) LREFL: This logical variable is used to tell the code whether or not the reflected field from the reflector surface is desired in the pattern computations when LGTD=true. The reflected field from the reflector surface can be obtained by setting LGTD=true, LREFL=true, LMP=false and LSR=false.

  g) THETAX: This real variable, input in degrees, is used as a criterion for switching from AI to GTD for both far field and near field calculations. If the field point angle $|\theta|$<THETAX, AI is used; otherwise, GTD is used. If THETAX is input as zero or, if no value is input (TO: Command is not used), it will be calculated as follows:

  $$THETAX=\theta_x = \sin^{-1} \frac{1}{\sqrt{A_w}}$$

  where $A_w$ is the aperture width in the specific pattern cut.

  h) ZXP: This real variable is input in the units specified by the variable IUNIT in the DG: Command. It is used as a range criterion only for near field calculations. If the range R (LRANG=true) or the distance from the aperture Z (LRANG=false) is less than ZXP, only GTD is used. Otherwise, the near field point angle is compared with THETAX to determine if AI or GTD is used.

  The default value of ZXP is zero; thus only AI is used inside the projected aperture unless ZXP is input by the TO: Command.

19-4

NOTE: When near field points are calculated within a few aperture
      widths in front of the aperture, it is recommended to compare
      both AI and GTD by using the above input parameters. However, AI
      calculations may not be accurate for points at wide angles
      outside the projected aperture, or for points closer than two
      aperture widths. GTD calculations may not be acccurate inside or
      close to the projected aperture for points farther than a few
      aperture widths, if there are "holes" or extreme variations in
      the aperture field. The reason for this is that GTD contains
      only the information for the direct geometrical optics field and
      the diffracted fields from the rim.

The usage of the above criteria for AI/GTD switching is summarized in the block diagram below:

BLOCK DIAGRAM FOR SWITCHING CRITERIA
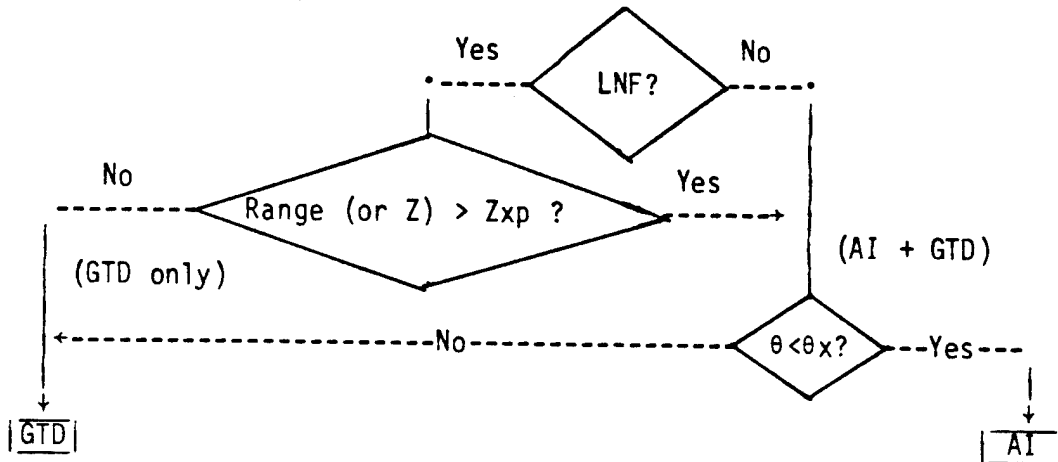


DIAGRAM FOR AI AND GTD REGIONS
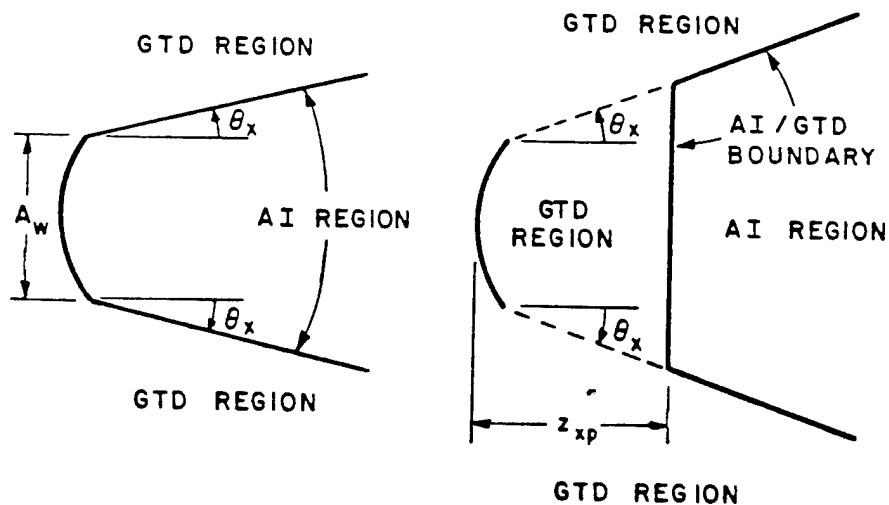


a) AI/GTD boundary of the default case (input $\theta_x=0.$, $Z_{xp}=0.$)

b) Non-zero $Z_{xp}$ and/or $\theta_x$ can be used to decide the AI/GTD boundary.

19-6

6. Read:  LMP,LSR,THEXF,THEXR

This read statement is used to choose either the Multi-Point GTD or the Segmented-Rim GTD for specified regions.

a) LMP:  This logical variable is used to enable the multi-point GTD.  (see Table 1 for details)

b) LSR:  This logical variable is used to enable the segmented rim GTD.  (see Table 1 for details)

c) THEXF:  This real variable ($\theta_{XF}$) specifies the switching angle from multi-point GTD to segmented rim GTD in the <u>front axis region</u> if both LMP and LSR are true.  If $\theta_{XF}$ is input as zero or no value is input (TO:  command is not used), it will be set to be 2°.

d) THEXR:  This real variable ($\theta_{XR}$) specifies the switching angle from multi-point GTD to segmented GTD in the <u>rear axis region</u> if both LMP and LSR are true.  If $\theta_{XR}$ is input as zero or no value is input (TO:  command is not used), it will be set to 5°.


NOTE:  If there are more than 9 diffraction points found by multi-point GTD (LMP=true), the code will be terminated unless the segmented-rim GTD is also used (i.e., LSR=true).

TABLE 1

THE USE OF LMP AND LSR
TO DETERMINE GTD MODELS USED FOR THE GTD REGION

| LMP | LSR | DECISION |
|-----|-----|----------|
| F | F | No diffracted field is desired. Spill-over and/or G.O. fields can be obtained if desired. |
| T | F | Diffracted field is calculated using the multi-point GTD alone. Input for $\theta_{XF}$ and $\theta_{XR}$ are neglected. |
| F | T | Diffracted field is calculated using the segmented rim GTD alone. Input for $\theta_{XF}$ and $\theta_{XR}$ are neglected. |
| T | T | Diffracted field is calculated using either the multi-point GTD or segmented rim GTD as specified by $\theta_{XF}$ and $\theta_{XR}$ (see Figure 1). |

REGION USING
MULTI-POINT GTD

REFLECTOR

REGION USING
SEGMENTED
RIM GTD

$\theta_{XR}$

$\theta_{XF}$

Z

REGION USING
SEGMENTED
RIM GTD

REGION USING
MULTI-POINT GTD
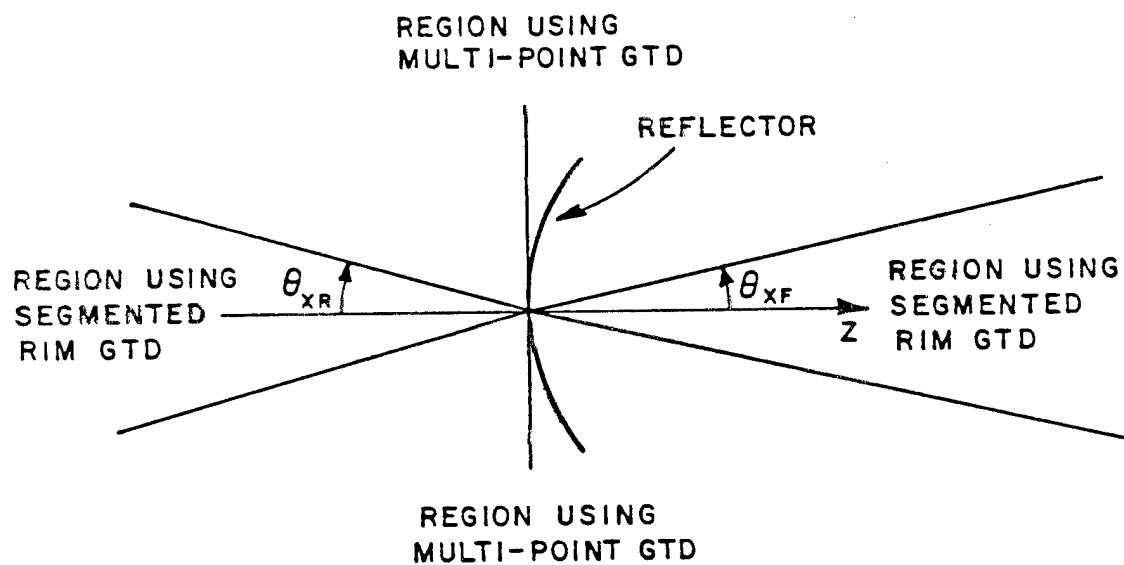
Figure 1.  Geometry for switching angles $\theta_{XF}$ and $\theta_{XR}$.

---
Command TT:   SPECIAL TEST OPTIONS
---

     This command enables the user to run certain infrequently used
tests.  CAUTION MUST BE USED IN SELECTING THE VALUES FOR THE FOLLOWING
VARIABLES!  VALUES SHOULD USUALLY BE IN THE RANGE SPECIFIED.


BLOCK DIAGRAM FOR CERTAIN TESTS


```
                    | Command TT: |     .
                           |
                  | READ:  LRIML,RIML |
                           |
                    |Read:  LTHEB|
                           |
                    | Read:  EPS |
                           |
               | Read:  LAPIN,LAPOUT |
                     .     |
                    | Read:  LCONVR |
                           |
               | Read:  LSFSC,LSFNRM |
                           |
              | Read:  LELPAT,NELPAT |
```
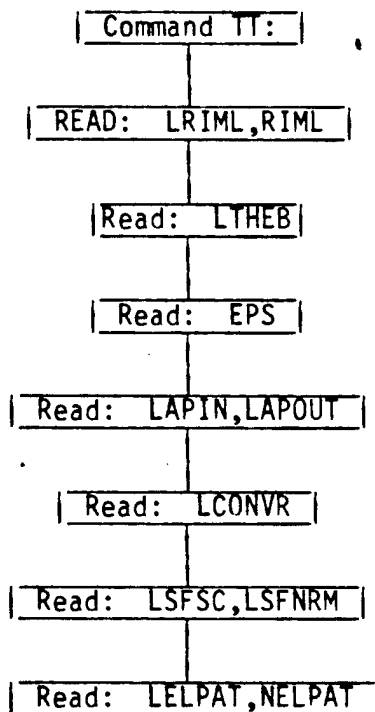
1. Read:  LRIML,RIML

   a) LRIML:  This logical variable is used to control the maximum
      length RIML of each segment on the reflector rim.  Normally,
      LRIML is false and RIML is calculated by the code itself.  For
      test purposes, LRIML can be set true; then RIML is read by the
      next input.  (normally set false)

   b) RIML:  If LRIML is true, this real variable is used to determine
      the maximum length (in units) of each rim segment for
      segmented-rim GTD for far field calculation.  For near field
      calculation, the maximum size is halved, i.e., twice many rim
      segments are used.  If LRIML is false, this input is ignored by
      the code.

2. Read: LTHEB

   a) LTHEB: This logical variable is used to decide if the shadow
      effect of the reflector surface on the diffracted field from the
      opposite side of the reflector rim should be included. If LTHEB
      is true, as that in default data, this shadow effect is
      included. This may cause discontinuities in the pattern because
      double diffraction is not included. (normally set true)

3. Read: EPS

   a) EPS: This real variable is used to define the shadow boundary
      region for the reflected field from the reflector in the near
      field GTD case. EPS is the distance in wavelength from the
      shadow boundary. For those points inside this region, the near
      field values are calculated by linear interpolation from the
      adjacent points just outside this region. Presently, EPS=0.,
      unless this command is used.

4. Read: LAPIN,LAPOUT

   This statement gives the option to store the aperture fields. If
   the same reflector system is used, one can store the aperture fields
   on the first run and then use the stored aperture fields for the
   next run.

   a) LAPIN: This logical variable enables one to read the stored
      aperture fields from Unit #20. If LAPIN is true, the stored
      aperture fields are read.

   b) LAPOUT: This logical variable enables one to store the aperture
      fields. If LAPOUT is true, the aperture fields are stored in
      Unit #20.

      Note that usually LAPIN and LAPOUT are set false because the
      aperture fields will take a great deal of computer storage space.

5. Read: LCONVR

This logical variable is used to determine whether the output field data will be converted to principal and cross polarization for far-field or near-field constant range case. If LCONVR=F, the output field data will be in $\theta$ and $\phi$ components. Normally, LCONVR=T.

6. Read: LSFSC,LSFNRM

This statement is used for surface perturbation (SP) or surface distortion (SD) cases.

a) LSFSC: This logical variable specifies whether the scattered field due to surface error is calculated or not. If LSFSC=T, the scattered field will be calculated. If LSFSC=F, the total field from the distorted reflector will be calculated. Normally, LSFSC=F.

b) LSFNRM: This logical variable specifies whether the surface error is given along the reflector surface normal or along the z-axis. If LSFNRM=T, the surface error input in SP: or SD: Command is along the reflector surface normal. If LSFNRM=F, the surface error is along z-axis.

7. Read: LELPAT,NELPAT

This statement is used to define the element pattern for Aperture Integration. The element pattern is given as $(\cos(\theta/2))^{NELPAT}$.

a) LELPAT: This logical variable enables one to change the element pattern. The default of LELPAT is false.

b) NELPAT: This integer variable specifies the power of $\cos(\theta/2)$ to determine the element pattern. Normally, NELPAT=2. If both AI and GTD are used (LAIC=T and LGTD=T in TO: ) to calculate reflector patterns, NELPAT will be set to 1 to make better switchover between AI and GTD unless LELPAT=true.

```
------------------------------------------------------
```
Command TE:  TEST APERTURE FIELD
```
------------------------------------------------------
```

This command enables the user to test the aperture field data at
any point on the aperture.


1.  Read:  LTEA, NTEA

    a) LTEA:  This logical variable is used to indicate whether a test
       for aperture field data is desired.

    b) NTEA:  This integer variable is used to specify the number of
       aperture field points.


2.  Read:  XXEA(N), YYEA(N)

    This read statement is executed NTEA times.

    a) XXEA(N):  This dimensional real variable specifies the x
       coordinate (in units) of the aperture field point to be tested.

    b) YYEA(N):  This dimensional real variable specifies the y
       coordinate (in units) of the aperture field point to be tested.


                  BLOCK DIAGRAM FOR TEST APERTURE FIELD


                          | Command:  TE |
                                 |
                                 |
                       | Read:  LTEA, NTEA |
                                 |
                                 |
                     | Loop through each point |
                     |        N=1, NTEA        |
                                 |
                                 |
                     | Read:  XXEA(N), YYEA(N) |


                                21-1
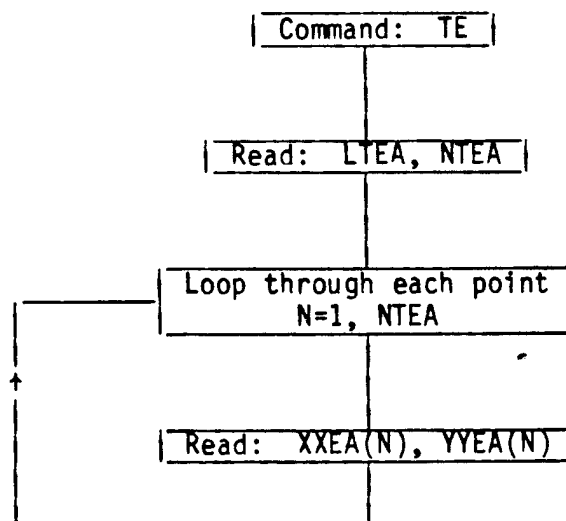
----------------------------------------------------------------
Command NF:  NEAR FIELD OUTPUT
----------------------------------------------------------------

This command enables the user to specify whether near field or far
field output is to be computed. It also specifies the PHI-plane cut and
coordinate origin for near field calculations as shown in Figure 1. The
units for the distance parameters are specified according to the value
of

$$IUNIT = \begin{cases} 1\text{->meters} \\ 2\text{->feet} \\ 3\text{->inches.} \end{cases}$$

The value of IUNIT is controlled by the DG: Command.

BLOCK DIAGRAM FOR NEAR FIELD/FAR FIELD OPTIONS

```
                        | Command NF: |
                              |
                 | Read:  (X00(I),I=1,3)|
                              |
                        | Read:  LNF |
                              |
       TRUE                 /     \                FALSE
    .------------------<  If(LNF)  >-----.
    |                      \     /        |
    | (Near Field)                        | (Far Field)
  | Read:  LRANG |                     | Read:  LRFCT |
       |                                      |
  | Read:  PHIE  |                            |
                             TRUE          /      \          FALSE
                          .-------------< If(LRFCT) >---.
                          |                \      /       |
                          |                              |
                   | Read: RANG |              | Range factor |
                                              | is suppressed |
```

22-1

1. READ:  (XOO(I),I=1,3)
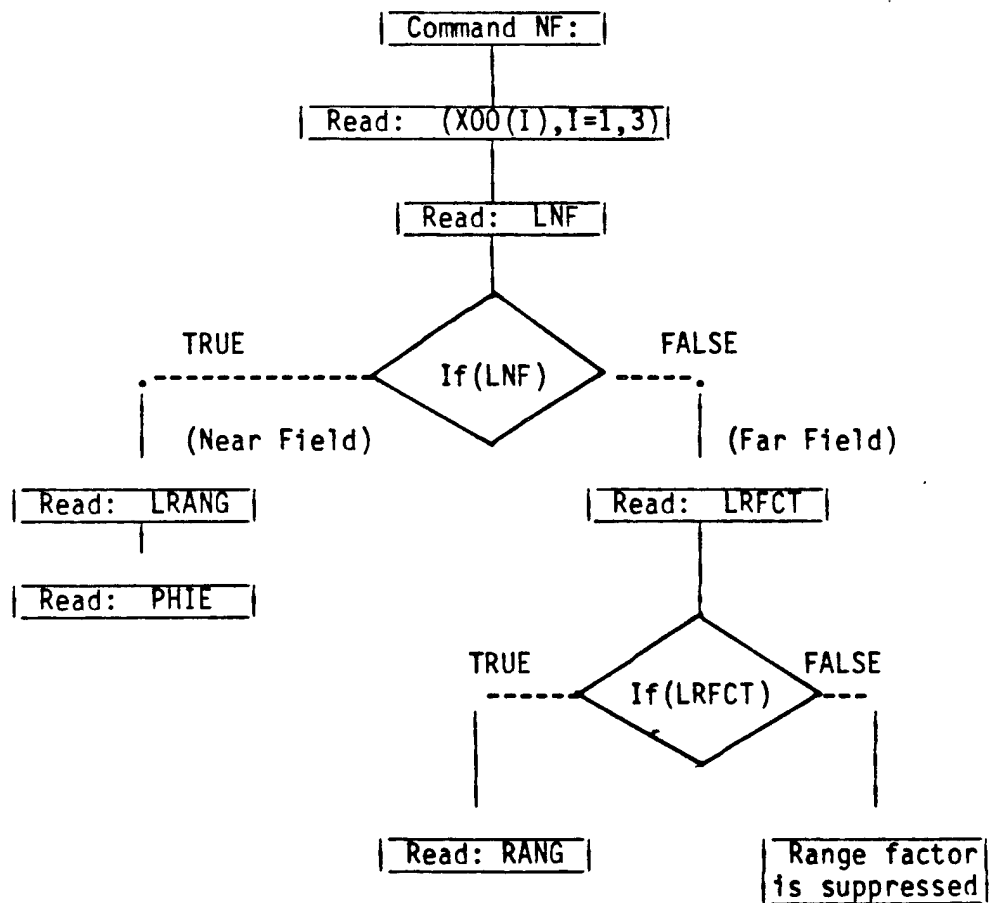
   a) XOO(I):  This dimensioned real variable is used to specify the
      origin of the coordinate system for near field observation
      points.  See Figure 1.  It is also used to specify a phase
      reference for the far field pattern.  If the NF:  Command is not
      used the default value of XOO = (0,0,ZOP) is used, where ZOP is
      the distance from the reflector vertex to the aperture plane.


2. READ:  LNF

   a) LNF:  This logical variable is used to tell the code whether or
      not near field output is to be computed.  If set false, far field
      patterns are computed.


3. READ:  LRFCT

   This statement is skipped if LNF=true.

   a) LRFCT:  This logical variable specifies whether the range factor
      is to be included for far field output.  If set false, the range
      factor is suppressed.


4. READ:  RANG

   This statement is skipped if LNF=true or LRFCT=false.

   a) RANG:  This real variable defines the far field range R at which
      the antenna fields are to be calculated as shown in Figure 1a.
      It is used only when LRFCT=true and LNF=false.


5. READ:  LRANG

   This statement is skipped if LNF=false.

   a) LRANG:  This logical variable specifies whether results for a
      given PHI-plane cut (PHIE) are to be computed for constant range
      R or constant Z distance from the aperture plane, for near field
      output.  See Figure 1.


6. READ:  PHIE

   This statement is skipped if LNF=false.

   a) PHIE:  This real variable is input in degrees and defines the
      PHI-plane cut for near field output as shown in Figure 1b.

22-2
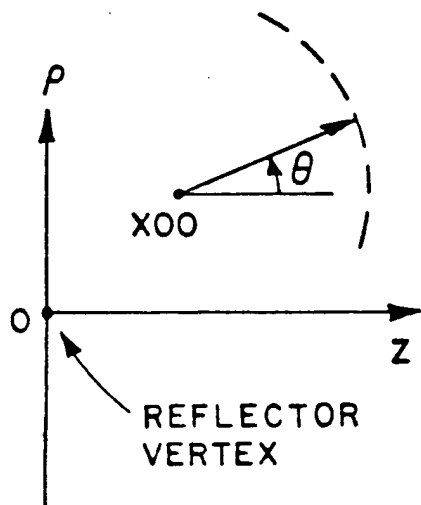
(a) Far Field (LNF=false)
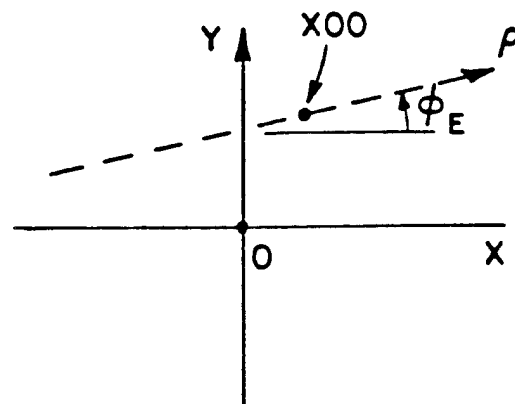
(b) PHIE-plane for Near Field (LNF=true)
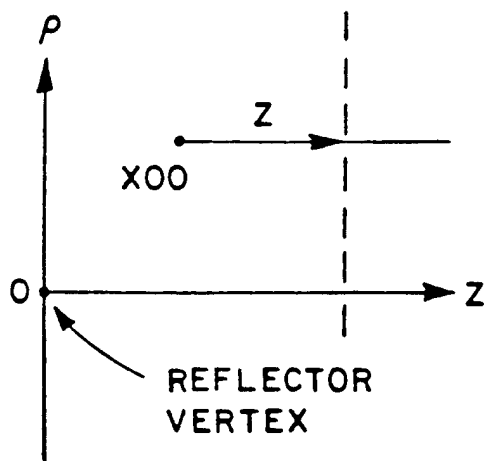
(c) Near Field, (LRANG=false)
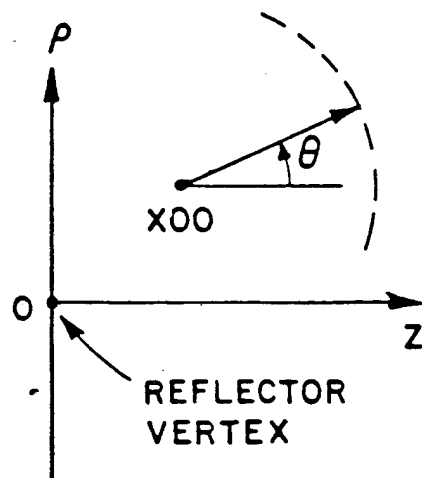
(d) Near Field, (LRANG=true)

Figure 1.  Coordinate systems for far field and near field pattern cuts.

EXAMPLE:

This example illustrates near field calculations with a constant z-cut. The reflector is a circular one with diameter D=10λ and an isotropic feed located at a focal distance F=10,000λ to simulate a planar aperture. The y component of the near field results as computed by AIC and GTD for the constant plane cut at Z=12λ are shown in Figures 2a and b, respectively. Good agreement is achieved by these two methods. The input data for AIC and GTD are given in Table 1 and Table 2, respectively.
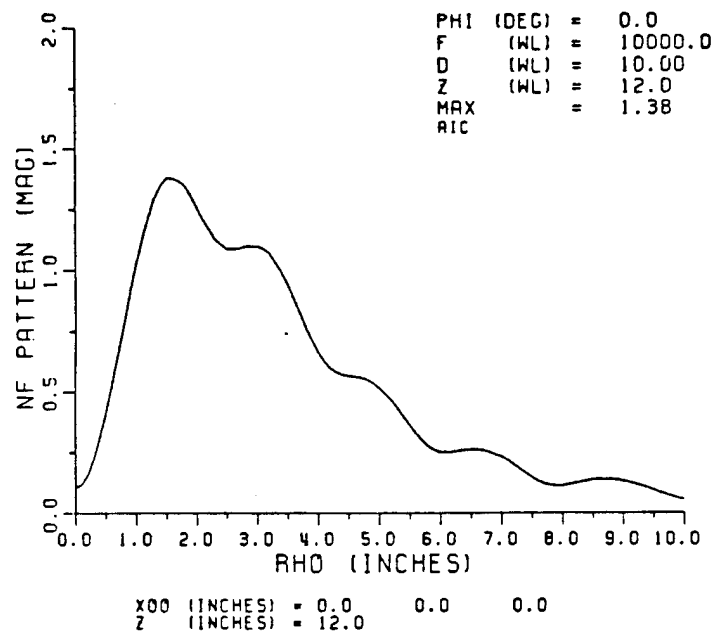
Figure 2a.   Near field pattern computed by OSU Reflector Antenna Code
             using AIC.
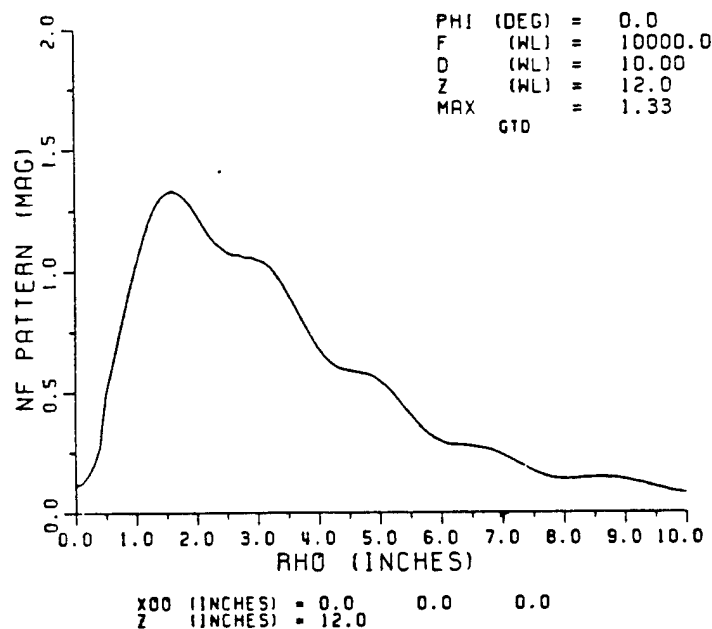


Figure 2b.   Near field pattern computed by OSU Reflector Antenna Code
             using GTD.

# TABLE 1

## INPUT DATA FOR NEAR FIELD CONSTANT
## Z-CUT PATTERN CALCULATION
## (AIC)

```
CM:        ***** PWSAI.DAT *****
CM:    NEAR-FIELD WITH CONSTANT Z-CUT
CE:          LAIC = T ONLY
DG:
1
3    10000.    0.2    0.2    10.    0
TO:
F    30.  1.
F    0    0    0    0
F    F    F    F    0
T .  T    F    0.8
T    F    F    F    F    F    0.    0.
F    F    0.   0.
FD:
0    T
F    0    F    1    90.   0.   F
2    0.   90.
0.   0.   0.   180.   0.   0.   180.
F    180.  180.
FQ:
1    11.81102362
NF:
0.   0.    0.
T
F
0.
PZ:
1
12.
0.    10.    0.1
F
PP:
1
1  1
2  1
XQ:
```

## TABLE 2

## INPUT DATA FOR NEAR FIELD CONSTANT
## Z-CUT PATTERN CALCULATION
## (GTD)

```
CM:   '   ***** PWSAI.DAT *****
CM:   NEAR-FIELD WITH CONSTANT Z-CUT
CE:          LGTD = T ONLY
DG:
1
3    10000.    0.2    0.2    10.    0
TO:
F   30.  1.
F   0    0    0    0
F   F    F    F    0
T.   T    F   0.8
F   F    F    F    T    T   0.   0.
T   T   0.  0.
TT:
F  0.
90.
0.04
F   F
F
F   F
F   0
FD:
0    T
F   0    F   1    90.   0.   F
2   0.   90.
0.   0.   0.   180.   0.   0.   180.
F   180.  180.
FQ:
1    11.81102362
NF:
0.   0.   0.
T
F
0.
PZ:
1
12.
0.   10.   0.1
F
PP:
1
1   1
2   1
XQ:
```

---
Command HZ: RADIATION HAZARD
---

This command enables the user to output radiation hazard data on the line printer. Regular pattern data is also output. The radiation hazard data includes power density in milliwatts per square centimeter, electric field in volts per meter and near field gain in dB.

1. READ: LRADHZ

   a) LRADHZ: This logical variable is used to tell the code whether or not radiation hazard data is to be computed. If set false, only regular pattern data is computed.

2. READ: PTRANS

   This statement is skipped if LRADHZ=false.

   a) PTRANS: This real variable defines the input power (in watts) to the reflector antenna, which is assumed to radiate all of this power.

BLOCK DIAGRAM FOR RADIATION HAZARD DATA

```
                    | Command HZ: |
                          |
                          |
              | Read:  LRADHZ          |
                          .
                                              FALSE
            <        If (LRADHZ)        >- - - - - - - - - -
                          |                                  |
                          |                                  |
                          | TRUE                             |
              | Read:  PTRANS |                              |
                          |                                  |
                          |_____|
                          |
                          |
                     | EXIT |
```
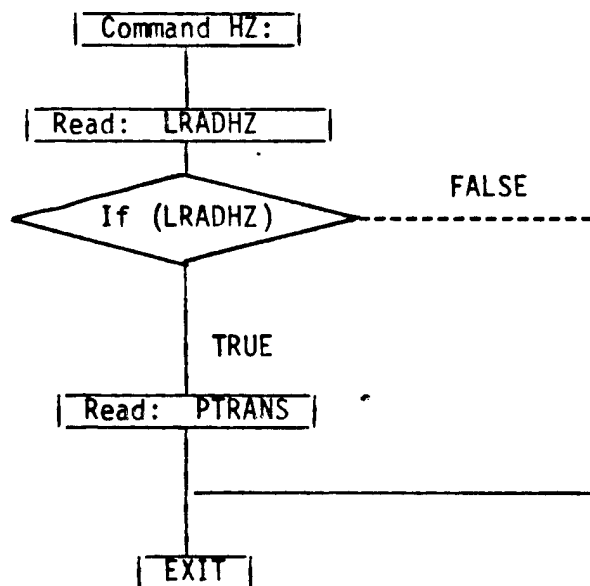
EXAMPLE:

This example uses a 24" circular reflector and default analytic feed to calculate the output for the radiation hazard applications. The input data are given in Table 1 while the output data are given in Table 2. The power density is calculated in milliwatts per square centimeter, and the electric field is calculated in volts per meter.

# TABLE 1

## INPUT DATA FOR CALCULATION OF RADIATION HAZARD

```
CM:        ***** C24HZ.DAT *****
CM: GENERAL EXAMPLE OF 24" CIRCULAR REFLECTOR
CE:      NEAR FIELD , RADIATION HAZARD
NF:
0.   0.   4.5
T
T
0.
HZ:
T
1.
PZ:
3
42.9497 107.3742 1073.742
0.   180.   10.
F
XQ:
```

# TABLE 2

## OUTPUT DATA FOR THE RADIATION HAZARD CALCULATION

```
************************************************************************
*                                                                    *
*   XQ:                                                              *
*                                                                    *
*                                                                    *
*                                                                    *
*                                                                    *
*                                                                    *
*        FREQUENCY =    11.000 GHZ                                    *
*                                                                    *
*        WAVELENGTH =    0.027273 METERS                             *
*        * THE FOLLOWING DIMENSION UNITS ARE IN WAVELENGTHS *        *
*        * ANGLES & PHASE ARE IN DEGREES *                           *
*                                                                    *
*                                                                    *
*        APERTURE DIAMETER =    22.35 WAVELENGTHS                    *
*                                                                    *
*        NUMBER OF RIM SEGMENTS=188                                  *
*                                                                    *
*                                                                    *
*        FOCAL DISTANCE =      7.45                                  *
*                                                                    *
*        GRIDX =   0.56     GRIDY =    0.56                          *
RGEOM:RHOS1,BOUND=   11.73480        13.42983                        
*                                                                    *
*     ZOP=   4.191000                                                *
*        DISTANCE FROM FOCUS TO RIM: RO=    11.642                   *
*                                                                    *
*        FEED LOCATED AT (      0.00,     0.00,     7.45,)           *
*                   ( REFERRED TO VERTEX )                           *
*                                                                    *
*                                                                    *
*        FEED POWER: PRAD = 0.138E+01                                *
*                                                                    *
*        NUMBER OF PRINCIPAL GRID LINES: MMAX= 41      NMAX= 43      *
*                                                                    *
*        APERTURE CENTER AT (   0.000,  0.000,       4.191 )         *
*                                                                    *
*                                                                    *
*                                                                    *
*                                                                    *
*                                                                    *
*        CPU TIME FOR APERTURE FIELD CALCULATION =  10.54 SECONDS    *
*                                                                    *
*                                                                    *
*        THE REFLECTED SHADOW BOUNDARIES IN THE PHIE PLANE ARE AT    *
*                                                                    *
*              RHOS1 =  -11.175     AND RHOS2 =   11.175             *
*                                                                    *
*        NEAR FIELD GAIN REFERENCE POINT:                            *
*              XREF = (   0.000,  0.000,        4.191)               *
*        NEAR FIELD GAIN REFERENCE POINT:                            *
*              XNREF= (   0.000,  0.000,       44.191)               *
*                                                                    *
*        RANGE FOR NEAR FIELD GAIN: RANFWL = 0.4000E+02              *
*                                                                    *
*        NEAR FIELD GAIN REF =    24.182                             *
*                                                                    *
*        SHADOW BOUNDARY ANGLES: TH1 =  253.74     TH2 =  106.26     *
*        THEB (DEG) =    90.000                                      *
```

TABLE 2 - CONTINUED

```
AI/GTD SWITCHOVER PARAMETERS:
   THETAX =  12.22      ZX =         0.000
     NT   =  19      P3X =    12.219
   NGTD1=   0      PG1I=     0.000
   NAI  =   3      PAI =     0.000
   NGTD2=  16      PG2I=    30.000

PHI =    0.00

   NEAR FIELD WITH CONSTANT RANGE R =          40.00:
                             40.00 FROM APERTURE
```

|  | PRINCIPAL POL | | | CROSS POL | | |
|---|---|---|---|---|---|---|
| THETA | MAG | DB | PHASE | MAG | DB | PHASE |
| 0.00 | 0.101E+01 | 24.29 | -33.8 | 0.766E-07 | -118.13 | -33.9 |
| 10.00 | 0.347E+00 | 14.99 | 163.5 | 0.263E-07 | -127.43 | 163.9 |
| 20.00 | 0.512E-01 | -1.62 | 23.5 | 0.366E-08 | -144.54 | 17.9 |
| 30.00 | 0.272E-01 | -7.12 | -54.8 | 0.201E-07 | -129.77 | 169.1 |
| 40.00 | 0.179E-01 | -10.75 | 172.6 | 0.120E-07 | -134.27 | 100.6 |
| 50.00 | 0.156E-01 | -11.96 | 9.3 | 0.136E-07 | -133.17 | -12.8 |
| 60.00 | 0.181E-01 | -10.64 | -162.5 | 0.608E-08 | -140.14 | 137.5 |
| 70.00 | 0.238E-01 | -8.27 | -3.3 | 0.712E-08 | -138.76 | 157.2 |
| 80.00 | 0.206E-01 | -9.55 | 169.2 | 0.625E-08 | -139.89 | 51.3 |
| 90.00 | 0.276E-01 | -7.01 | -43.2 | 0.320E-08 | -145.72 | 162.0 |
| 100.00 | 0.151E-01 | -12.26 | 128.9 | 0.260E-08 | -147.50 | 102.5 |
| 110.00 | 0.601E-02 | -20.24 | -159.6 | 0.523E-08 | -141.45 | -163.4 |
| 120.00 | 0.299E-02 | -26.32 | 156.0 | 0.549E-08 | -141.03 | 153.3 |
| 130.00 | 0.185E-02 | -30.50 | 1.8 | 0.568E-08 | -140.73 | -0.6 |
| 140.00 | 0.132E-02 | -33.42 | 121.2 | 0.998E-08 | -135.84 | 153.9 |
| 150.00 | 0.960E-03 | -36.17 | -176.7 | 0.605E-08 | -140.18 | -103.8 |
| 160.00 | 0.105E-02 | -35.36 | -162.7 | 0.269E-07 | -127.21 | 174.0 |
| 170.00 | 0.453E-03 | -42.70 | -137.8 | 0.144E-07 | -132.63 | 50.9 |
| 180.00 | 0.227E-01 | -8.71 | -61.5 | 0.383E-07 | -124.15 | 146.0 |

***** OUTPUT FOR RADIATION HAZARD *****

```
   TRANSMITTED POWER = 0.1000E+01 WATTS

   RANGE = 0.1091E+01 METERS
```

| THETA | E TOTAL (V/m) | POWER DENSITY (mW/cm**2) | NF GAIN (DB) |
|---|---|---|---|
| 0.00 | 0.8227E+02 | 0.1795E+01 | 24.29 |
| 10.00 | 0.2820E+02 | 0.2109E+00 | 14.99 |
| 20.00 | 0.4164E+01 | 0.4600E-02 | -1.62 |
| 30.00 | 0.2212E+01 | 0.1298E-02 | -7.12 |
| 40.00 | 0.1456E+01 | 0.5625E-03 | -10.75 |
| 50.00 | 0.1267E+01 | 0.4262E-03 | -11.96 |
| 60.00 | 0.1474E+01 | 0.5764E-03 | -10.64 |
| 70.00 | 0.1938E+01 | 0.9958E-03 | -8.27 |
| 80.00 | 0.1672E+01 | 0.7413E-03 | -9.55 |
| 90.00 | 0.2240E+01 | 0.1331E-02 | -7.01 |
| 100.00 | 0.1223E+01 | 0.3971E-03 | -12.26 |
| 110.00 | 0.4882E+00 | 0.6323E-04 | -20.24 |
| 120.00 | 0.2426E+00 | 0.1561E-04 | -26.32 |
| 130.00 | 0.1500E+00 | 0.5966E-05 | -30.50 |
| 140.00 | 0.1071E+00 | 0.3045E-05 | -33.42 |
| 150.00 | 0.7804E-01 | 0.1615E-05 | -36.17 |
| 160.00 | 0.8567E-01 | 0.1947E-05 | -35.36 |
| 170.00 | 0.3680E-01 | 0.3592E-06 | -42.70 |
| 180.00 | 0.1841E+01 | 0.8994E-03 | -8.71 |

TABLE 2 - CONTINUED

```
*
*
*
*       THE REFLECTED SHADOW BOUNDARIES IN THE PHIE PLANE ARE AT
*
*               RHOS1 -  -11.175     AND RHOS2 -   11.175
*
*       NEAR FIELD GAIN REFERENCE POINT:
*               XREF - (   0.000,    0.000,          4.191)
*       NEAR FIELD GAIN REFERENCE POINT:
*               XNREF- (   0.000,    0.000,        104.192)
*
*       RANGE FOR NEAR FIELD GAIN: RANFWL = 0.1000E+03
*
*       NEAR FIELD GAIN REF -     32.141
*
*       SHADOW BOUNDARY ANGLES: TH1 -  253.74     TH2 -  106.26
*       THEB (DEG) -   90.000
*
*       AI/GTD SWITCHOVER PARAMETERS:
*       THETAX -  12.22     ZX -         0.000
*         NT    -  19       P3X -    12.219
*         NGTD1-   0        PG1I-     0.000
*         NAI  -   2        PAI -     0.000
*         NGTD2-  17        PG2I-    20.000
*
*   PHI -    0.00
*
*     NEAR FIELD WITH CONSTANT RANGE R -            100.00:
*                                   100.00 FROM APERTURE
*
W                        PRINCIPAL POL                CROSS POL
W
W    THETA         MAG        DB      PHASE    MAG        DB       PHASE
*
W     0.00     0.800E+00    30.20    -22.8  0.606E-07  -112.21    -22.9
W    10.00     0.281E-01     1.11     17.3  0.214E-08  -141.26     16.7
W    20.00     0.134E-01    -5.29     13.7  0.643E-08  -131.69   -168.8
W    30.00     0.100E-01    -7.86    -72.0  0.287E-08  -138.69     86.6
W    40.00     0.913E-02    -8.65    173.7  0.174E-08  -143.06     29.4
W    50.00     0.909E-02    -8.69     17.7  0.230E-08  -140.63    151.1
W    60.00     0.731E-02   -10.58   -156.1  0.209E-08  -141.44    -96.6
W    70.00     0.918E-02    -8.60     27.5  0.205E-08  -141.64   -144.2
W    80.00     0.852E-02    -9.25   -164.5  0.193E-08  -142.14     69.9
W    90.00     0.104E-01    -7.52    -21.8  0.124E-08  -146.00    145.9
W   100.00     0.739E-02   -10.48    161.3  0.130E-08  -145.59    117.6
W   110.00     0.324E-02   -17.66    -91.4  0.214E-08  -141.25    -96.0
W   120.00     0.160E-02   -23.76    -79.0  0.252E-08  -139.85    -82.0
W   130.00     0.975E-03   -28.08   -166.9  0.265E-08  -139.39   -169.3
W   140.00     0.714E-03   -30.78     13.4  0.511E-08  -133.69      0.3
W   150.00     0.593E-03   -32.40    126.9  0.655E-08  -131.54    141.3
W   160.00     0.409E-03   -35.63    163.6  0.548E-08  -133.08    105.2
W   170.00     0.331E-03   -37.46   -125.1  0.117E-07  -126.51    -70.6
W   180.00     0.105E-01    -7.46    -94.3  0.705E-07  -110.90    137.0
W
    ***** OUTPUT FOR RADIATION HAZARD *****
W
W       TRANSMITTED POWER - 0.1000E+01 WATTS
W
W       RANGE - 0.2727E+01 METERS
W
W    THETA        E TOTAL         POWER DENSITY        NF GAIN
W                 (V/m)           (mW/cm**2)           (DB)
W
W     0.00     0.6498E+02        0.1120E+01           30.20
W    10.00     0.2281E+01        0.1380E-02            1.11
```

TABLE 2 - CONTINUED

| W | | | | | | W |
|---|---|---|---|---|---|---|
| W | 20.00 | 0.1093E+01 | 0.3168E-03 | -5.29 | | W |
| W | 30.00 | 0.8127E+00 | 0.1752E-03 | -7.86 | | W |
| W | 40.00 | 0.7420E+00 | 0.1460E-03 | -8.65 | | W |
| W | 50.00 | 0.7385E+00 | 0.1447E-03 | -8.69 | | W |
| W | 60.00 | 0.5938E+00 | 0.9353E-04 | -10.58 | | W |
| W | 70.00 | 0.7463E+00 | 0.1477E-03 | -8.60 | | W |
| W | 80.00 | 0.6921E+00 | 0.1271E-03 | -9.25 | | W |
| W | 90.00 | 0.8450E+00 | 0.1894E-03 | -7.52 | | W |
| W | 100.00 | 0.6008E+00 | 0.9575E-04 | -10.48 | | W |
| W | 110.00 | 0.2630E+00 | 0.1835E-04 | -17.66 | | W |
| W | 120.00 | 0.1302E+00 | 0.4499E-05 | -23.76 | | W |
| W | 130.00 | 0.7919E-01 | 0.1664E-05 | -28.08 | | W |
| W | 140.00 | 0.5805E-01 | 0.8938E-06 | -30.78 | | W |
| W | 150.00 | 0.4818E-01 | 0.6156E-06 | -32.40 | | W |
| W | 160.00 | 0.3322E-01 | 0.2928E-06 | -35.63 | | W |
| W | 170.00 | 0.2690E-01 | 0.1920E-06 | -37.46 | | W |
| W | 180.00 | 0.8504E+00 | 0.1918E-03 | -7.46 | | W |

THE REFLECTED SHADOW BOUNDARIES IN THE PHIE PLANE ARE AT

RHOS1 = -11.175     AND RHOS2 =   11.175

NEAR FIELD GAIN REFERENCE POINT:
        XREF = (   0.000,    0.000,          4.191)
NEAR FIELD GAIN REFERENCE POINT:
        XNREF= (   0.000,    0.000,       1004.203)

RANGE FOR NEAR FIELD GAIN: RANFWL = 0.1000E+04

NEAR FIELD GAIN REF =     52.141

SHADOW BOUNDARY ANGLES: TH1 = 253.74    TH2 = 106.26
THEB (DEG) =    90.000

AI/GTD SWITCHOVER PARAMETERS:
THETAX = 12.22     ZX =        0.000
    NT  = 19      P3X =   12.219
    NGTD1=   0     PG1I=    0.000
    NAI  =   2     PAI =    0.000
    NGTD2=  17     PG2I=   20.000

PHI =    0.00

NEAR FIELD WITH CONSTANT RANGE R =        1000.01:
                              1000.01 FROM APERTURE

| | | PRINCIPAL POL | | | CROSS POL | | | |
|---|---|---|---|---|---|---|---|---|
| W | THETA | MAG | DB | PHASE | MAG | DB | PHASE | W |
| W | 0.00 | 0.138E+00 | 34.93 | 27.8 | 0.105E-07 | -107.47 | 27.8 | W |
| W | 10.00 | 0.127E-02 | -5.75 | 159.3 | 0.996E-10 | -147.89 | 158.5 | W |
| W | 20.00 | 0.273E-03 | -19.14 | 33.7 | 0.627E-09 | -131.92 | 11.2 | W |
| W | 30.00 | 0.679E-03 | -11.22 | -60.3 | 0.340E-09 | -137.23 | -93.2 | W |
| W | 40.00 | 0.666E-03 | -11.39 | 178.2 | 0.313E-09 | -137.95 | 148.8 | W |
| W | 50.00 | 0.860E-03 | -9.17 | 34.9 | 0.194E-09 | -142.09 | -92.1 | W |
| W | 60.00 | 0.810E-03 | -9.69 | -146.6 | 0.167E-09 | -143.40 | -44.3 | W |
| W | 70.00 | 0.896E-03 | -8.81 | 40.5 | 0.182E-09 | -142.68 | -113.7 | W |
| W | 80.00 | 0.870E-03 | -9.06 | -152.9 | 0.175E-09 | -143.00 | 73.4 | W |
| W | 90.00 | 0.985E-03 | -7.99 | -12.3 | 0.131E-09 | -145.50 | 134.4 | W |
| W | 100.00 | 0.812E-03 | -9.67 | 174.8 | 0.140E-09 | -144.94 | 120.5 | W |
| W | 110.00 | 0.378E-03 | -16.30 | -62.2 | 0.155E-09 | -144.04 | -67.3 | W |
| W | 120.00 | 0.189E-03 | -22.35 | -23.6 | 0.250E-09 | -139.89 | -26.7 | W |

TABLE 2 - CONTINUED

| W | | | | | | | | W |
|---|---|---|---|---|---|---|---|---|
| W | 130.00 | 0.114E-03 | -26.71 | -77.2 | 0.379E-09 | -136.30 | -66.5 | W |
| W | 140.00 | 0.830E-04 | -29.48 | 137.7 | 0.474E-09 | -134.34 | 118.8 | W |
| W | 150.00 | 0.712E-04 | -30.81 | -76.9 | 0.678E-09 | -131.23 | -71.5 | W |
| W | 160.00 | 0.435E-04 | -35.08 | -12.6 | 0.389E-09 | -136.06 | -75.0 | W |
| W | 170.00 | 0.431E-04 | -35.16 | 73.9 | 0.136E-08 | -125.17 | 116.7 | W |
| W | 180.00 | 0.113E-02 | -6.82 | 103.8 | 0.741E-09 | -130.47 | -83.8 | W |
| W | | | | | | | | W |

**** OUTPUT FOR RADIATION HAZARD ****

TRANSMITTED POWER = 0.1000E+01 WATTS

RANGE = 0.2727E+02 METERS

| THETA | E TOTAL (V/m) | POWER DENSITY (mW/cm**2) | NF GAIN (DB) |
|---|---|---|---|
| 0.00 | 0.1121E+02 | 0.3332E-01 | 34.93 |
| 10.00 | 0.1035E+00 | 0.2843E-05 | -5.75 |
| 20.00 | 0.2216E-01 | 0.1303E-06 | -19.14 |
| 30.00 | 0.5516E-01 | 0.8070E-06 | -11.22 |
| 40.00 | 0.5411E-01 | 0.7767E-06 | -11.39 |
| 50.00 | 0.6987E-01 | 0.1295E-05 | -9.17 |
| 60.00 | 0.6584E-01 | 0.1150E-05 | -9.69 |
| 70.00 | 0.7281E-01 | 0.1406E-05 | -8.81 |
| 80.00 | 0.7073E-01 | 0.1327E-05 | -9.06 |
| 90.00 | 0.8001E-01 | 0.1698E-05 | -7.99 |
| 100.00 | 0.6600E-01 | 0.1156E-05 | -9.67 |
| 110.00 | 0.3074E-01 | 0.2506E-06 | -16.30 |
| 120.00 | 0.1532E-01 | 0.6228E-07 | -22.35 |
| 130.00 | 0.9276E-02 | 0.2282E-07 | -26.71 |
| 140.00 | 0.6746E-02 | 0.1207E-07 | -29.48 |
| 150.00 | 0.5783E-02 | 0.8870E-08 | -30.81 |
| 160.00 | 0.3537E-02 | 0.3319E-08 | -35.08 |
| 170.00 | 0.3506E-02 | 0.3260E-08 | -35.16 |
| 180.00 | 0.9155E-01 | 0.2223E-05 | -6.82 |

CPU TIME = 45.03 SECONDS

***** OUTPUT LISTING FROM THE OSU REFLECTOR ANTENNA CODE ( NASL4 ) *****

---------------------------------------------------------------
    Command PZ:   OUTPUT PATTERN POINTS
---------------------------------------------------------------

    This command enables the user to specify the output data.  For far
field patterns (LNF=false) this command specifies the PHI-plane pattern
cuts and the initial, final and incremental values for the theta pattern
angle.  For near field computations this command specifies the
constant-range cuts (LRANG=true) or the constant-Z cuts (LRANG=false)
for which the fields are to be computed.  The output pattern parameters
are shown in Table 1.  The units for the distance parameters are
specified according to the value of

$$
IUNIT \quad = \quad
\begin{array}{l}
1\text{->meters} \\
2\text{->feet} \\
3\text{->inches.}
\end{array}
$$

The value of IUNIT is controlled by the DG: Command.


               BLOCK DIAGRAM FOR OUTPUT PATTERN CUTS



                            24-1
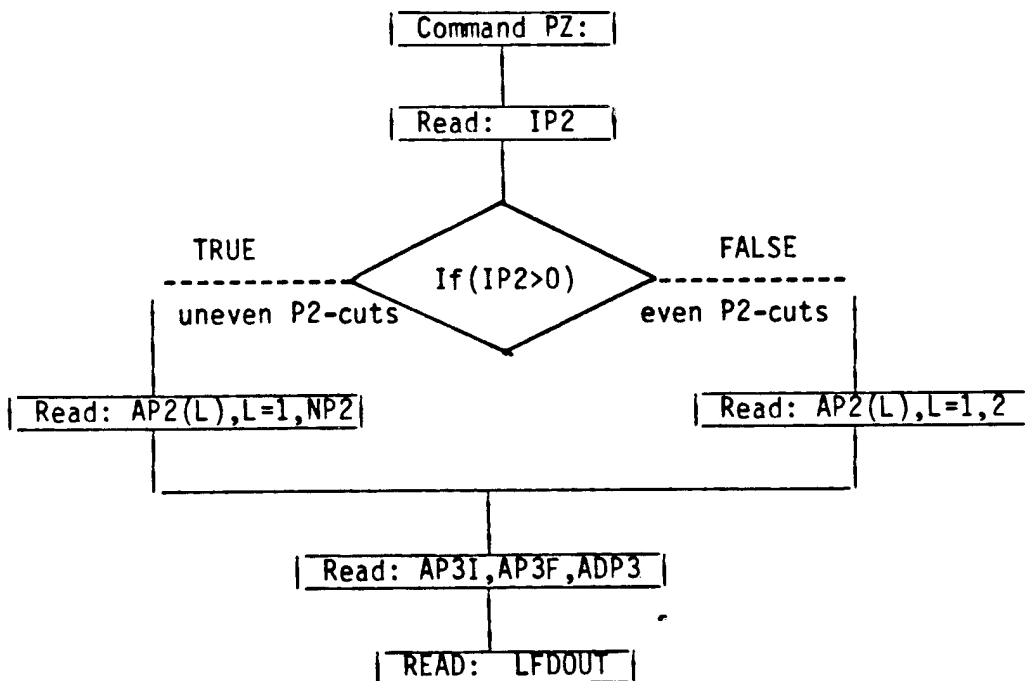
The following read statements control the output pattern parameters by use of the variables P2 and P3 as given in Table 1.

1.  READ:  IP2

    a) IP2:  This integer variable is used to specify the number of pattern cuts for the output data for each frequency.  Its absolute value NP2=|IP2| is the number of pattern cuts to be calculated.  If IP2 is positive (IP2>0), unevenly spaced-increments can be used as follows.  Presently 1<|IP2|<10.

2.  READ:  (AP2(L),L=1,NP2)

    a) AP2(L):  This dimensioned real variable defines the Lth value of P2 for output pattern data.  This read statement is used only for IP2>0.

3.  READ:  (AP2(L),L=1,2)

    This read statement is used for negative values of IP2(IP2<0); then, evenly spaced increments are used as follows:

    a) AP2(L):  This is a dimensioned real variable.  AP2(1) is the initial value of P2 for the output pattern data.  AP2(2) is the P2 increment for the output pattern data.

4.  READ:  AP3I,AP3F,ADP3

    a) AP3I:  This real variable defines the initial value of P3 for each pattern cut.  AP3I>-180 degrees for far field (LNF=false).

    b) AP3F:  This real variable defines the final value of P3 for each pattern cut.  AP3F<+180 degrees for far field (LNF=false).

    c) ADP3:  This real variable defines the value by which P3 is to be incremented for the output pattern.  ADP3>0.

Two examples of pattern cuts are shown in Figure 1.  A constant range case (LRANG=true) is shown in Figure 1a for which a full 360 pattern cut is calculated for either far field, (LNF=false), or near field, (LNF=true) by input of AP3I=-180, and AP3F=180.  A constant-Z example is illustrated in Figure 1b for near field (LNF=true) in which AP3I, and AP3F are input in the units specified by IUNIT in the DG:  Command.

## TABLE 1

## USE OF VARIABLES P2 AND P3

| | P2 | P3 |
|---|---|---|
| Input Variables | AP2(L) | AP3I,AP3F,ADP3 |
| Far Field (LNF=false) | $\phi$ | $\theta$ |
| Near Field* <br> Constant-R (LRANG=true) | R | $\theta$ |
| Near Field* <br> Constant-Z (LRANG=false) | Z | RHO |

*For near field computations the $\phi$-plane cut is defined by PHIE as specified by the NF: Command.
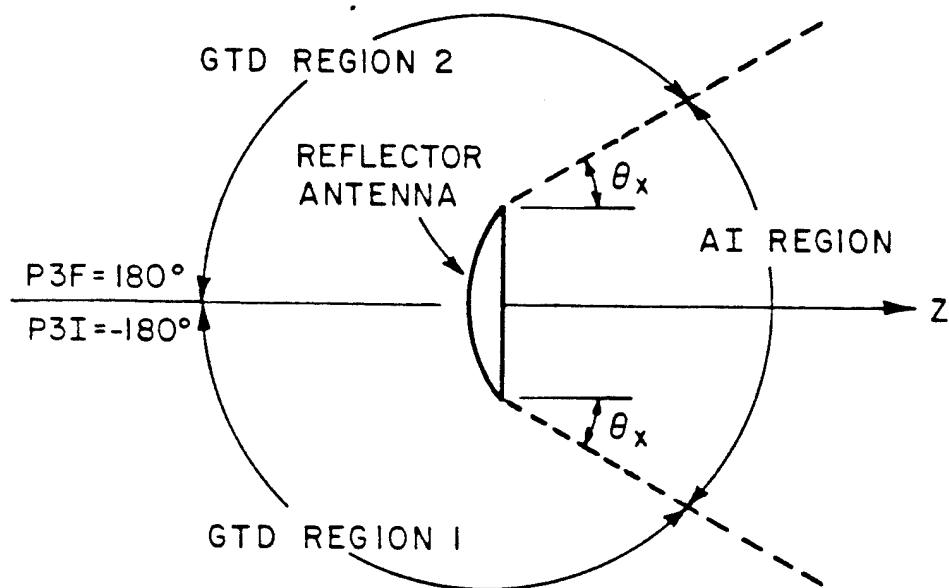
a) Constant range (LRANG=true)

Figure 1. Types of Output Pattern Cuts.
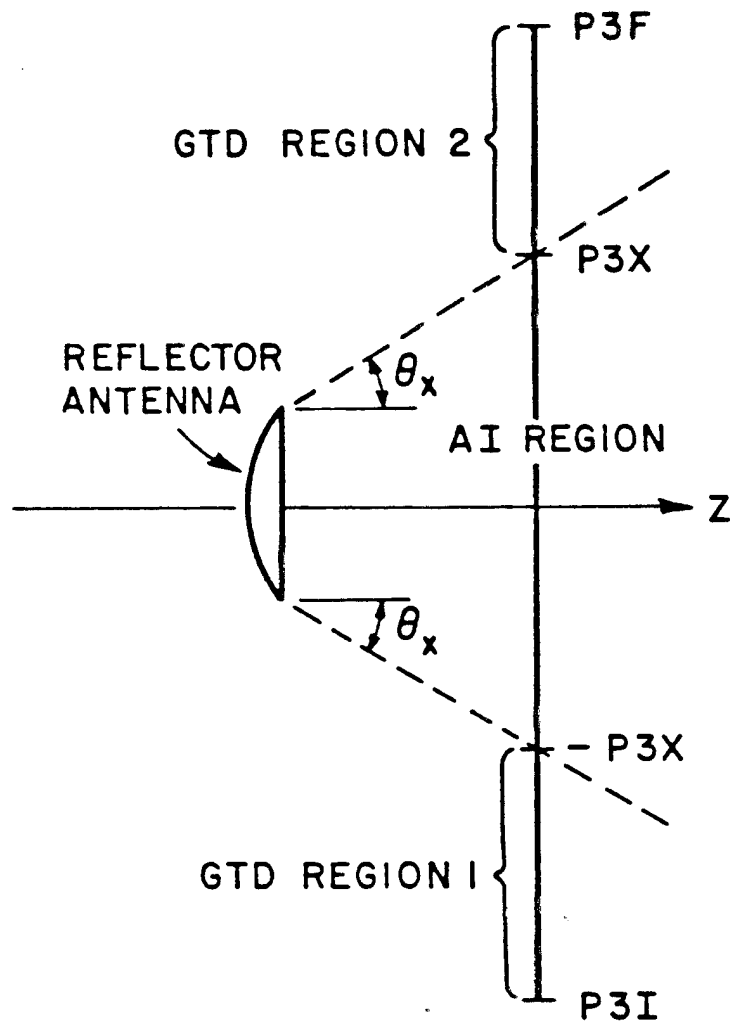
b) Constant Z (LRANG=false)

Figure 1. Continued.

5. Read: LFDOUT

   b) LFDOUT: This logical variable is used to output pattern data on
      Unit #7 with the same format as that used for feed pattern data
      in the FD: Command (with LDB=true). Thus, feed pattern data can
      be directly generated from this code by using one of the
      following commands: the AP:, WG:, or BR: Command. Two
      separate runs are needed: one to generate the feed pattern, the
      second to calculate the reflector pattern. (normally set false)


   NOTE: When LFDOUT=true is used to generate the feed data for the
         next run, the relationship between P2 ($\phi$) and PHIN ($\phi_n$) which
         will be used for the FD: Command in the subsequent run, must
         be noticed for feed patterns which are unsymmetric with
         respect to y-axis (ISYM=0 or |ISYM|=2). This relationship
         results from the difference between the coordinate systems in
         the two different runs which is described in Figure 2 where
         X-Y is the coordinate system of the reflector for the present
         run and x-Y is the coordinate system of the feed for the next
         run. In other words, the directions of the reflector and
         feed axes are reversed. If |ISYM|=1 or 3, this relationship
         can be ignored because of symmetry in the feed patterns.
         Some samples are also shown in Table 2.

24-6
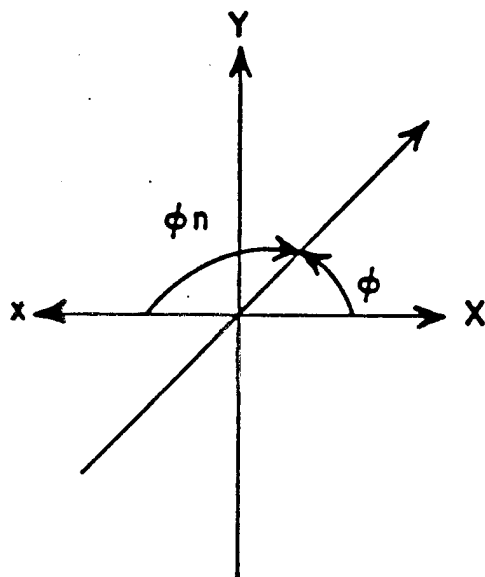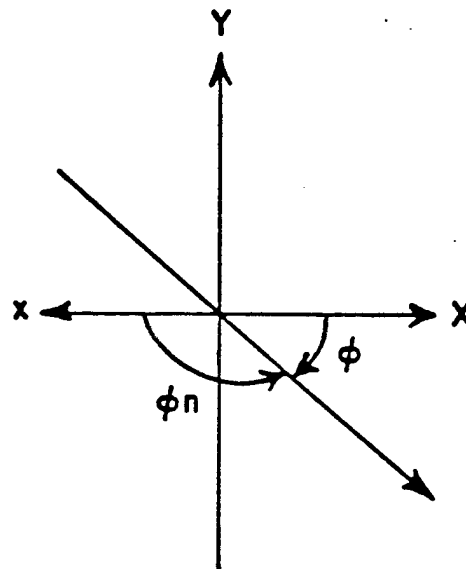
$\phi<\phi, \quad \phi_n=180-\phi$          $\phi<\phi, \quad \phi_n=-180-\phi$

Input Angle ($\phi$):   P2 (in the current PZ:   Command)

Output feed pattern angle ($\phi_n$):   PHIN (to be used in the subsequent FD: Command)

Figure 2.   Relationship between $\phi$ and $\phi_n$.

## TABLE 2

### SAMPLE INPUT ANGLES FOR DESIRED FEED PATTERN ANGLES

| Desired Feed Angle | Required Input Angle |
|---|---|
| PHIN($\phi_n$) | P2($\phi$) |
| -180. | 0. |
| -135. | - 45. |
| - 90. | - 90. |
| - 45. | -135. |
| 0. | 180. |
| 45. | 135. |
| 90. | 90. |
| 135. | 45. |
| 180. | 0. |

---
Command PP:  PLOT RADIATION PATTERN
---

This command enables the user to specify the plotted data for output patterns of the reflector antenna fields.  Other commands are used to plot feed patterns (PF:  Command) and Y-integration plots (PY: Command).  All plot data for reflector antenna fields are output on Unit #3.


BLOCK DIAGRAM FOR PLOTS OF OUTPUT PATTERNS


```
                          |Command PP:|
                               |
                               |
                          |Read:   NMO|
                               |
                               |
        ------------------|DO M=1,NMO|
       |                       |
       |                       |
       |              |Read:   MO(M),NPLT(M)|
       |                       |
       |                       |
       |      -------|DO NI=1,NPLT(M)|
       |     |                 |
       |     |                 |
       |     |        |Read:   NO(M,NI),IO(M,NI)|
       |     |                 |
       |     |                 |
        -----------------|CONTINUE|
```
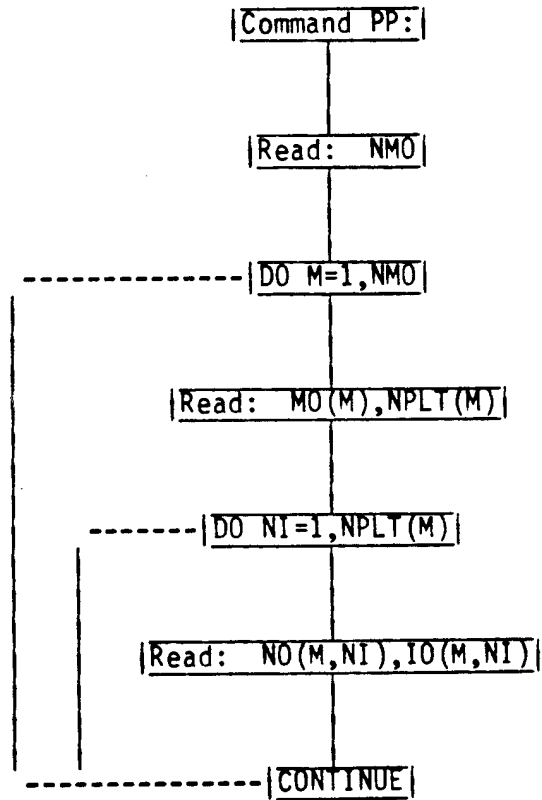
1.  READ:  NMO

    a) NMO:  This integer variable specifies the number of field
       terms for which output patterns are to be plotted.
       Typically, NMO=1, and MO(1)=1 in the next read statement to
       plot only the total field.  The value of NMO is independent
       of the number of patterns specified by IP2 in the PZ:
       Command; each pattern will be plotted with NMO=1.  Data for
       the plots are output on Unit #3.

2.  READ:  MO(M),NPLT(M)

    This read statement is executed NMO times.

    a) MO(M):  This integer specifies the field terms to be plotted
       as follows:

    MO =
    - 1 Total field
    - 2 Radiating field from the reflector surface
    - 3 Scattered field from feed blockage
    - 4 Scattered field from the reflector/strut scattering
    - 5 Scattered field from the feed/strut scattering
    - 6 Scattered field from both the reflector/strut and feed/strut scattering
    - 7 Scattered field from cracks

    b) NPLT(M):  This integer specifies the number of plots for
       each field term.

3.  READ:  NO(M,NI),IO(M,NI)

    This read statement is executed NPLT(M) times for each field
    term.

    a) NO(M,NI):  This integer specifies the polarization component
       to be plotted as follows:

    NO =
    - 1 X component (for LRANG=F); or principal polarized component (for LRANG=T)
    - 2 Y component (for LRANG=F); or cross polarized component (for LRANG=T)
    - 3 Z component (for LRANG=F; for near field only)

    b) IO(M,NI):  This integer specifies the format to be plotted
       as follows:

    IO =
    - 1 magnitude of a field term.
    - 2 dB value of a field term.
    - 3 phase of a field term in degrees.

```
----------------------------------------------------------
```
Command LP:   LINE PRINTER LISTING
```
----------------------------------------------------------
```

This command enables the user to specify whether a line printer listing of the results is desired.  It sets a flag so that data will be written out on a line printer.  Line printer data is output on Unit #6.


   1.  READ:   NPRI

      a) NPRI:  This integer variable specifies the number of printouts and controls the format of the printout as summarized below:

| | DATA OUTPUT ON LINE PRINTER | |
|---|---|---|
| | Near field with constant-Z plane cut (LRANG=F) | Far field and constant range near field (LRANG=T) |
| NPRI | X,Y,Z components | Principal and cross polarized components |
| -1 | -------------------- | Field terms for M = 1, 2, 3, 6 |
| 1 | Total field in magnitude, dB and phase forms. | Total field in magnitude dB and phase forms. |
| 2 or larger | In addition to the total field, the field terms indexed by M=JP(I), I=2, NPRI are printed out in the same format as the total field.  (See M values below) | The field terms indexed by M=JP(I), I=1,NPRI are printed out (See M values below) |

NO =

1 Total field
2 Radiating field from the reflector surface
3 Scattered field from feed blockage
4 Scattered field from the reflector/strut scattering
5 Scattered field from the feed/strut scattering
6 Scattered field from <u>both</u> the reflector/strut and feed/strut scatter<span style="text-decoration: overline">ing</span>
7 Scattered field from cracks

2. READ; (JP(I),I=1,NPRI)

This statement is skipped if LNF=false or NPRI < 2. (LNF is read in NF:)

a) JP(I) The input values for this integer array specify the field terms to be printed out as shown in the table above.

------------------------------------------------------------
Command FB:  FEED BLOCKAGE
------------------------------------------------------------

This command enables the user to simulate feed blockage by a circular or rectangular flat plate.  The units for the distance parameters are specified by the value of the integer IUNIT in the DG: Command.

BLOCK DIAGRAM FOR FEED BLOCKAGE

```
                        | Command FB: |
                               |
        | Read:  NBK,(XBO(N),N=1,3)|
                               |
                                              YES
                      /               \
                     <    NBK=0?        >---------------
                      \               /                |
                               |                       |
                               |                       |
                              NO |                      |
                                              YES       |
                      /               \                 |
                     <    NBK=2?        >-------         |
                      \               /       |          |
                               |              |          |
                                     | Read: AB,BB |      |
                                     | for rectangular|   |
                              NO |                      |
                  | Read: AB    |                       |
                  | for circular|                       |
                        |                               |
                         ------------->| <--------
                                       
                              | EXIT |
```

27-1
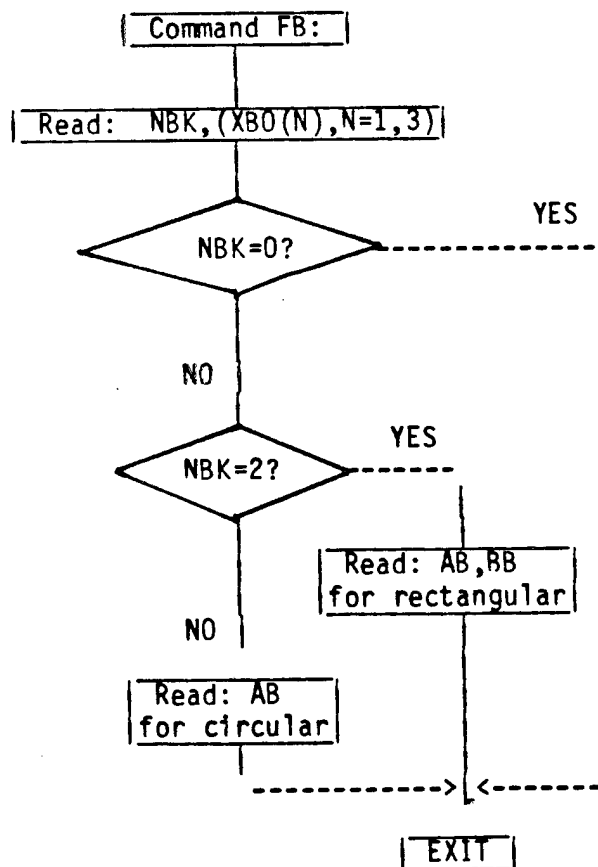
1. READ:  NBK,(XBO(N),N=1,3)

    a) NBK:  This integer variable is used to specify the feed blockage type as shown in Figure 1.

        NBK = 0 No feed blockage

        NBK = 1 Circular feed blockage

        NBK = 2 Rectangular feed blockage

    b) XBO(N):  This dimensioned real variable is used to specify the position (in rectangular coordinates) of the center of the feed blockage plate, relative to the reflector vertex as shown in Figure 2.

2. READ:  AB

    a) AB:  This real variable is read when circular feed blockage is desired.  It specifies the diameter of the plate.

3. READ:  AB,BB

These real variables are read when rectangular feed blockage is desired.

    a) AB:  The width (in X) of the plate.

    b) BB:  The height (in Y) of the plate.

      NOTE:  Feed blockage (FB) is modeled by the far-field approximation of Physical Optics model of a rectangular or circular flat plate.  These models will be valid in the main beam and near side lobe region provided that the observation range is located at the far-field of these plates, i.e., the observation range (constant-R or constant-Z in NF:)  must be greater than or equal to $2D^2/\lambda$ where D is the maximum dimension of the feed blockage plates.

NBK = 1



NBK = 2
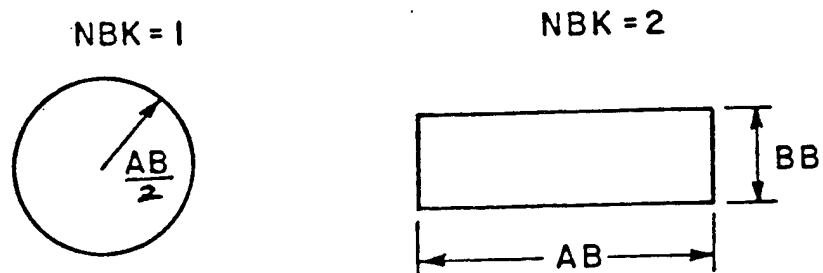
Figure 1. Feed blockage models.
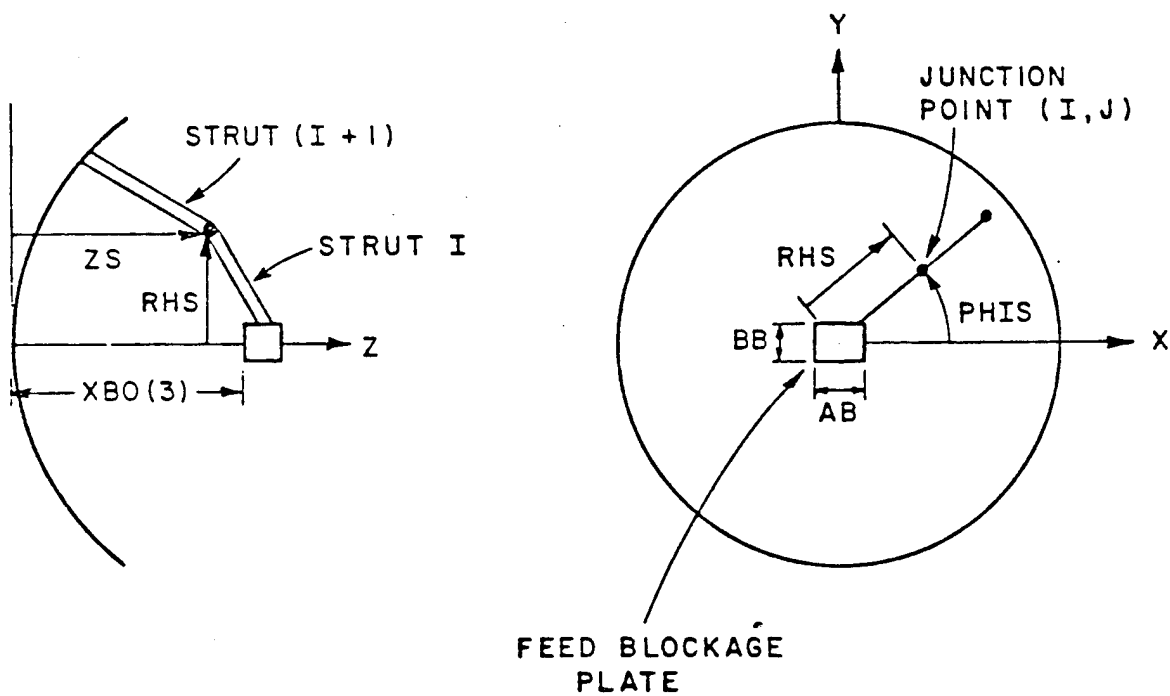


Figure 2. Strut/feed blockage geometries.

---
Command ST:  STRUT SCATTERING
---

This command enables the user to specify the geometry of the supporting struts of reflector antennas.  Two models of the supporting struts are used as shown in Figure 1.  For the model shown in Figure 1(a), referred to as reflector/strut scattering, the incident fields on the struts are the reflected fields from the reflector surface.  For the one shown in Figure 1(b), referred to as feed/strut scattering, the incident fields are from the primary feed.  Each strut can have either circular or general cross section and can be divided into small sections.  Each section will be considered as an individual strut inside the code and can be further subdivided into small segments.

The axis of each strut is straight; however, a piecewise linear strut can be modeled by joining several struts at their end-points.  In such a case, the coordinates of the end-point at each junction will be input as end-points for both adjacent struts.

1.  Read:  ISTRUT

    a) ISTRUT:  This integer variable specifies the type of strut scattering.

       ISTRUT = 0:  Reflector/strut scattering, the incident vectors on the strut are assumed to be parallel to the reflector $\hat{Z}$ axis.

       ISTRUT = 1:  Reflector/strut scattering, the incident vectors on the strut are found by a ray tracing technique.

       ISTRUT = 2:  Feed/strut scattering

       ISTRUT = 3:  Both reflector/strut and feed/strut scattering

2.  Read:  NS,GRST,THEST,LGEN,LBSAME,LCSAME,LNSAME

    a) NS:  This integer variable specifies the number of struts.  Presently, NS<108.

    b) GRST:  This real variable specifies the size of the segment used in subdividing a section of each strut.  The grid size GRST (input in units) controls the number of segments into which each section is subdivided, and should not exceed 0.5 wavelengths.  Presently, a maximum of 63 segments can be used on any strut section.
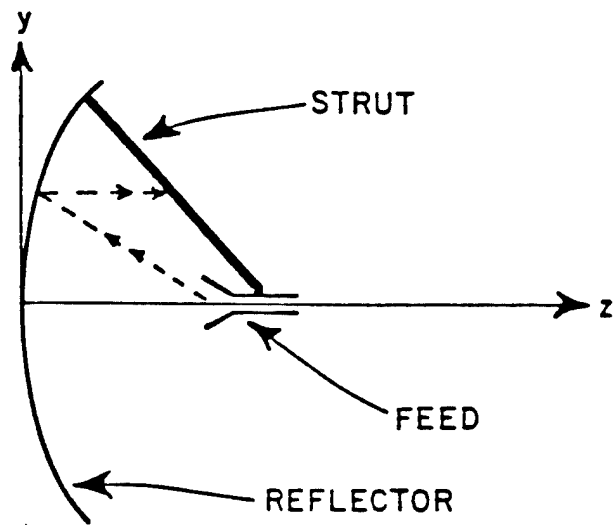
(a) reflector/strut scattering
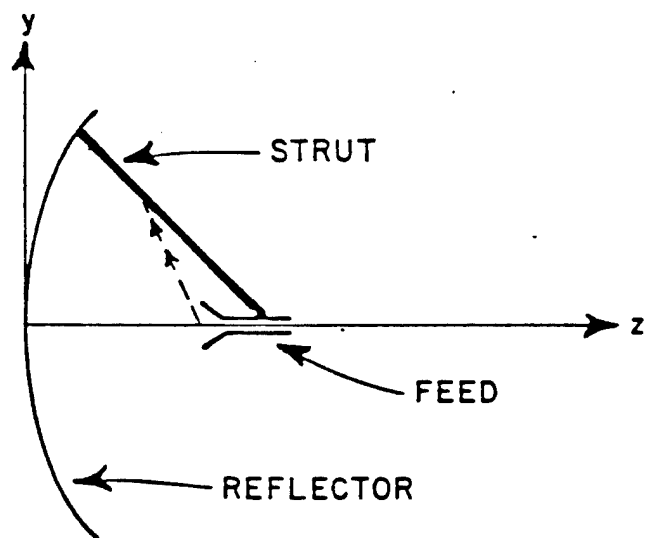


(b) feed/strut scattering

Figure 1. Geometry of a reflector with one strut.

c) THEST: This real variable is used to adjust the maximum theta angle for which strut scattering is included. Normally, 90<THEST<120°; other values may be used to approximate the shadowing of the strut scattering by the reflector surface. Presently, THEST=100°, unless this command is used.

d) LGEN: This logical variable specifies whether the struts have circular or general cross sections. The eigenfunction solution is used for circular cross sections whereas a moment method solution is used for general cross sections. Each strut can be subdivided into sections for the purposes of allowing sufficiently small segments, yet not exceeding the maximum number of segments per section. LGEN=F for struts with circular cross section.

e) LBSAME: This logical variable specifies whether the beta angle ($\beta$) for all struts is the same or not. The definition of $\beta$ is the angle between the incident vector and the unit vector along the axis of the strut at the mid-point of the strut as shown in Figure 2. The unit vector along the strut axis is directed from the first end-point of the strut to the second end-point.

f) LCSAME: This logical variable specifies whether all the struts have same cross section or not. If LCSAME is true, the strut cross section is input only once.

g) LNSAME: This logical variable specifies whether all struts are divided into same number of sections or not. (Note that each section can be further subdivided into small segments by the variable GRST).
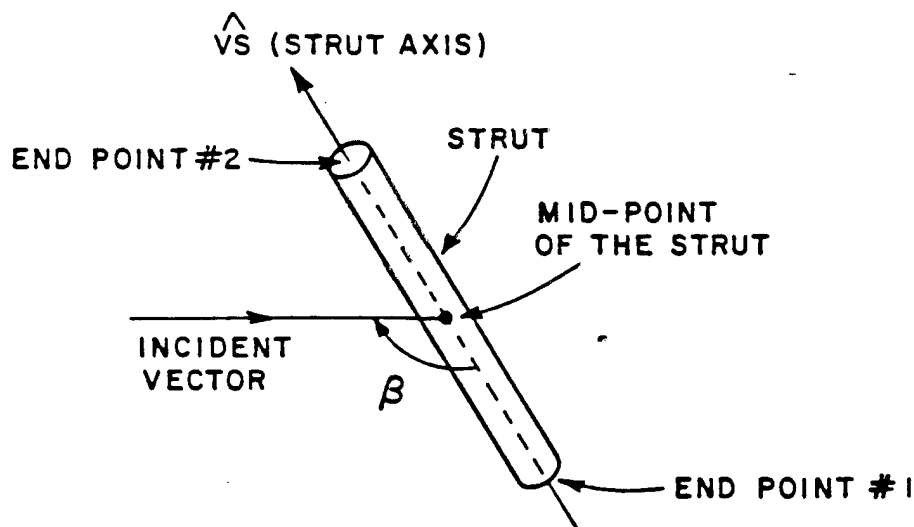


Figure 2. Geometry for the definition of $\beta$ angle.

28-3

3. Read: RHSU,PHIS(M,J),ZSU

This statement is executed twice for the end-points (J=1,2) for each strut (strut #M). The coordinate system is cylindrical, referred to the vertex of the reflector as shown in Figure 3.

a) RHSU: This real variable specifies the rho-coordinate of the $J^{th}$ end point on the $M^{th}$ strut.

b) PHIS(M,J): This is a real variable specifying the phi-coordinate (in degrees) of the end-points (J=1,2) on the $M^{th}$ strut.

c) ZSU: This is a real variable specifying the Z-coordinate of the $J^{th}$ end-point on the $M^{th}$ strut.

4. Read: NSS(M)

a) NSS(M): This integer variable specifies the number of sections into which strut M is to be subdivided. (This is useful for struts which are too long to be subdivided into sufficiently small segments because the number of segments would exceed that allowed by the dimensioned variables. Presently, 63 segments are allowed per section.)

If LNSAME=T, this variable is read only for the first strut (M=1). Note that the maximum number of subdivided sections for all the struts is 108. An example with 3 strut sections is given in Figure 4.

5. Read: DPU

This statement is executed only for struts with circular cross sections (LGEN=F). If all struts have the same cross section (LCSAME=T), this statement is executed only for the first strut. If LCSAME=F, this statement is executed for each strut.

a) DPU: This real variable specifies the diameter of the $M^{th}$ strut. Presently, the maximum strut diameter is 10 wavelengths.

28-4
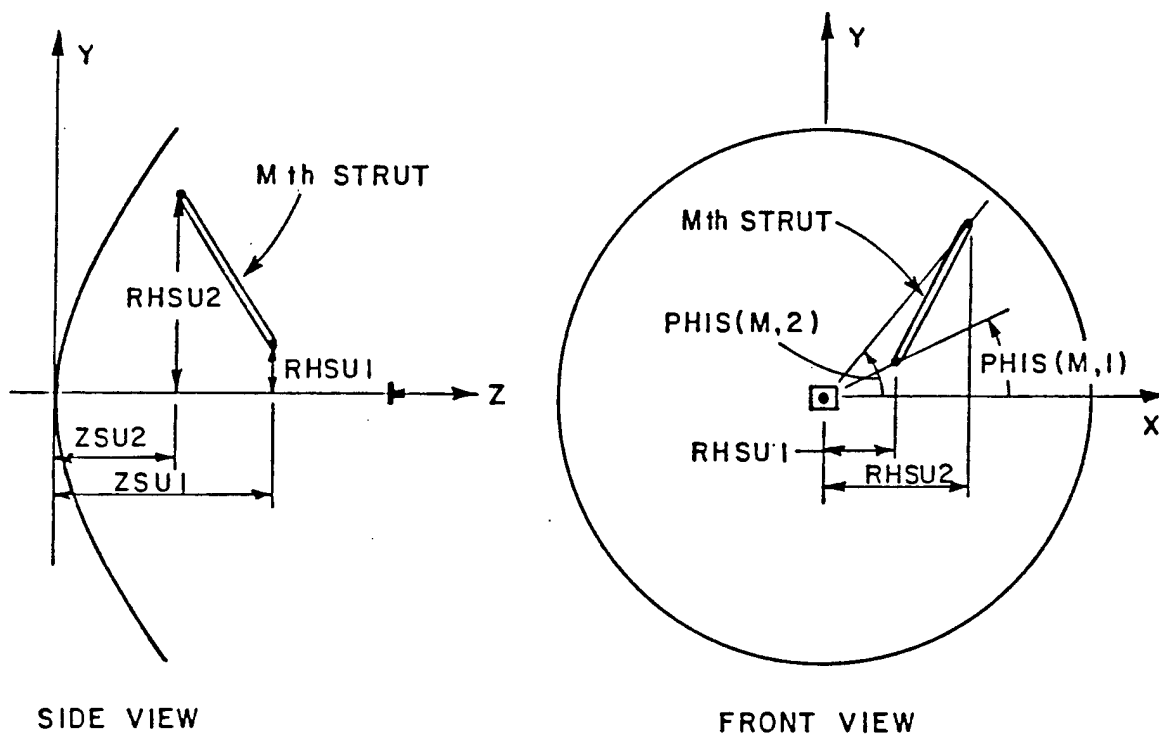
Figure 3. Coordinate system for the end points of the $M^{th}$ strut.

END POINT #2

STRUT #M

SECTION

GRST

SEGMENT

END POINT #1
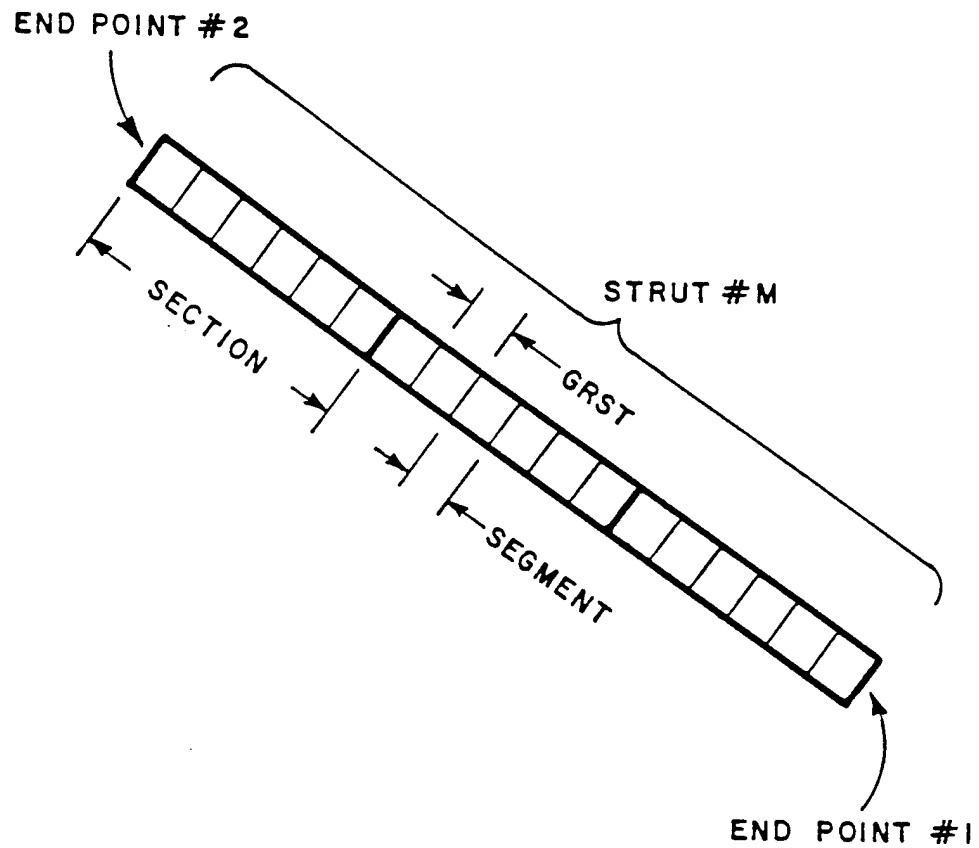
Figure 4. Example of a strut divided into 3 sections NSS(M)=3. In this particular example each section is divided into 6 segments by the choice of GRST.

The following statements are executed only for general struts (LGEN=T).

6. Read: TWIST2

   If LBSAME=F, this statement is executed for each strut. If LBSAME = T, this statement is executed only for the first strut.

   a) TWIST2: This variable specifies the twist angle of the strut with respect to the X'-axis (-Z axis of the refelctor coordinate system for $\beta=90°$) as shown in Figure 5.

7. Read: NM

   If LCSAME=F, this statement is executed for each strut; if LCSAME=T, this statement is executed for the first strut only.

   a) NM: This integer variable specifies number of points which describe the rim of the cross section of the strut. For struts with rectangular cross section, for example, NM should be at least equal to 4. Points on the line which connects two adjacent points can also be input as shown in Figure 5. A maximum of 10 points can be input for each strut cross section. The length between any two points should not exceed 0.25$\lambda$. The coordinates of these points are input in the next statement.

8. Read: XXCS(K), YYCS(K), K=1,NM

   This statement is executed when NM is input in the previous read statement.

   a) XXCS(K): This dimensioned variable specifies the x-coordinate of the $K^{th}$ point of the rim of the cross section in the local coordinate system of the strut.

   b) YYCS(K): This dimensioned variable specifies the y-coordinate of the Kth point of the rim of the cross section in the local coordinate system of the strut. An example of a rectangular strut is given in Figure 6.
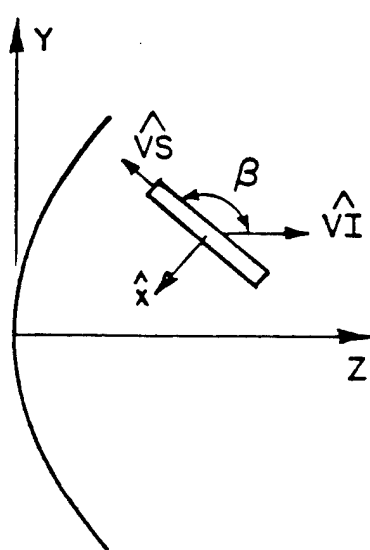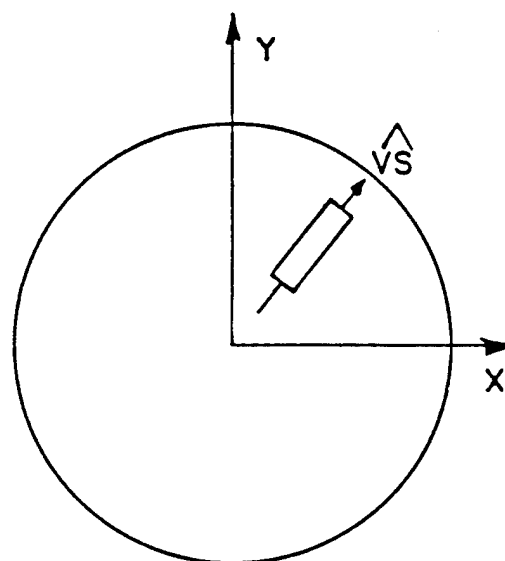
SIDE VIEW                 FRONT VIEW
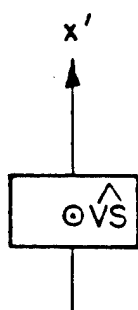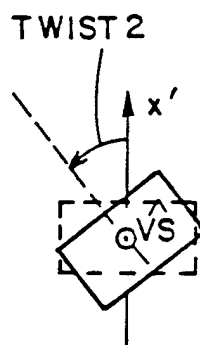
NO TWISTING              TWISTING

Figure 5. Definition of twist angle.

$\hat{VI}$: Incident unit vector

$\hat{VS}$: Unit vector along the strut axis

28-8
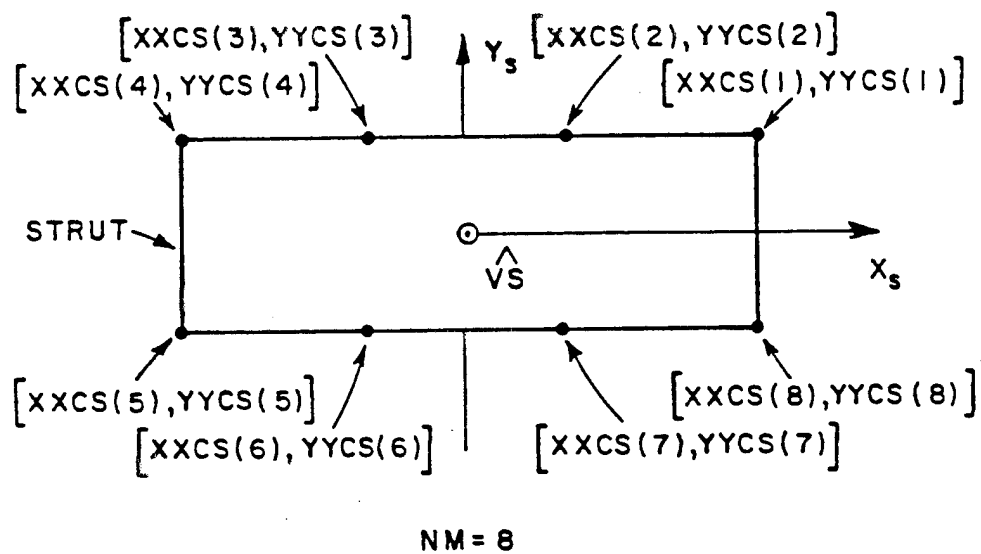
Figure 6.  Local coordinate system of a strut with rectangular cross section.

# BLOCK DIAGRAM FOR STRUT SCATTERING
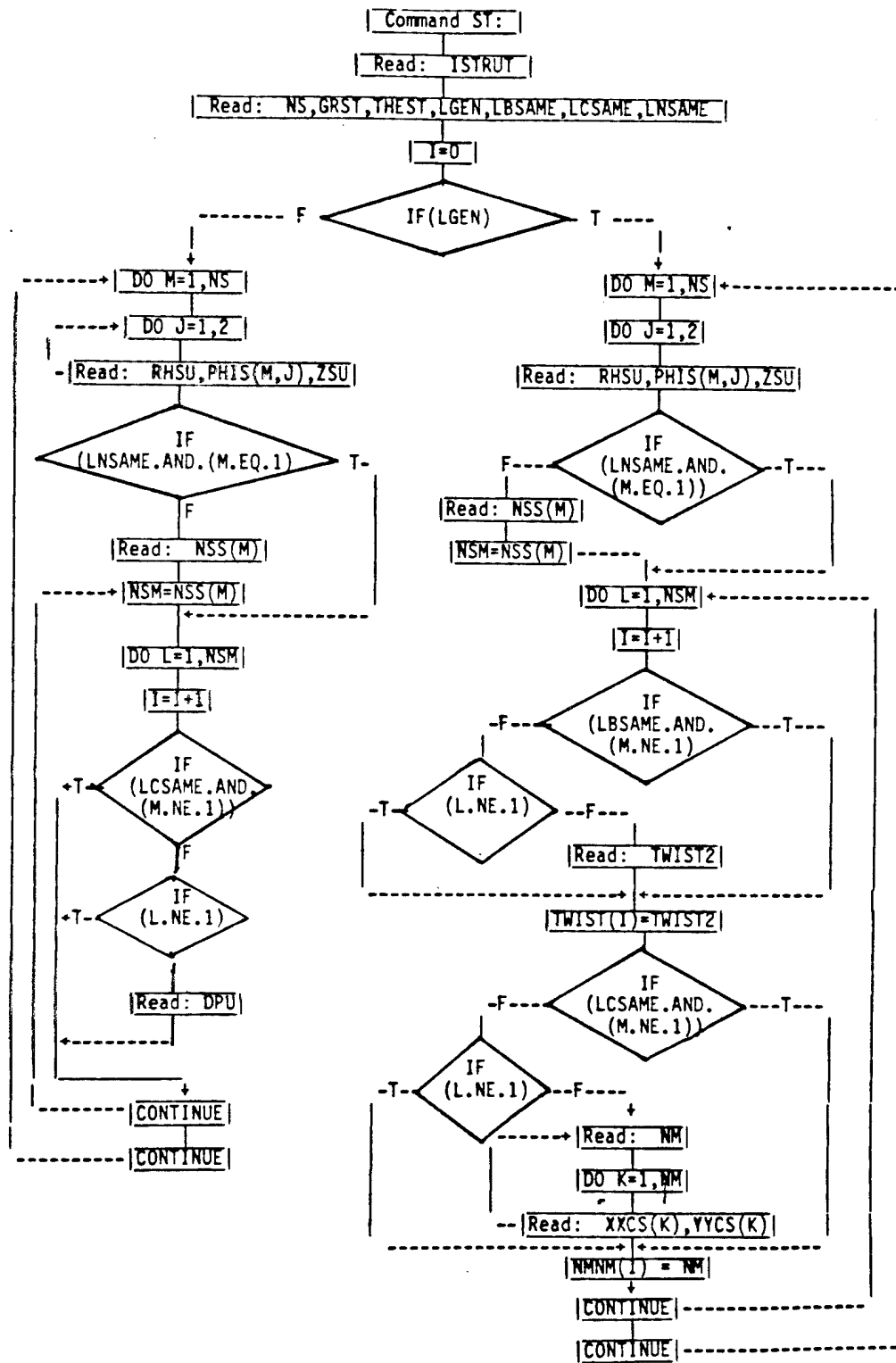
Example 1:

This example illustrates how struts with general cross sections (LGEN=T) are input. The default antenna with one rectangular strut is used and shown in Figure 7. The cross section of the strut is 0.75" x 0.43". The feed patterns are the same as in example 2 of Appendix B and the blockage effects of the feed are also included in the calculation. The 90° cut patterns of total field and reflector/strut scattering field (ISTRUT=1) are given in Figure 8. In Figure 8(b), the null at $\theta=-68.1°$ is due to the situation that the field point is right along the strut axis. The input data are given in Table 1.

# TABLE 1

## INPUT DATA FOR THE EXAMPLE WITH SINGLE RECTANGULAR STRUT

```
CM:        ***** STRUT1.DAT *****
CM: EXAMPLE OF REFLECTOR STRUT SCATTERING
CM:         1 RECTANGULAR STRUT
CE:      .         FEED BLOCKAGE
FD:
0    T
T    0    F    1    90.   0.    F
2    0.   90.
14
  .0.       1.0000    0.    0.    0.
  10.       0.9575    0.    0.    0.
  20.       0.8419    0.    0.    0.
  30.       0.6840    0.    0.    0.
  40.       0.5200    0.    0.    0.
  50.       0.3772    0.    0.    0.
  60.       0.2664    0.    0.    0.
  70.       0.1866    0.    0.    0.
  80.       0.1358    0.    0.    0.
  90.       0.10521   0.    0.    0.
 120.       0.03588   0.    0.    0.
 132.       0.05475   0.    0.    0.
 160.       0.01884   0.    0.    0.
 180.       0.02240   0.    0.    0.
   0.       1.0000    0.    0.    0.
  10.       0.9660    0.    0.    0.
  20.       0.8714    0.    0.    0.
  30.       0.7375    0.    0.    0.
  40.       0.5900    0.    0.    0.
  50.       0.4522    0.    0.    0.
  60.       0.3360    0.    0.    0.
  70.       0.2456    0.    0.    0.
  80.       0.1813    0.    0.    0.
  90.       0.13778   0.    0.    0.
 120.       0.09170   0.    0.    0.
 132.       0.07900   0.    0.    0.
 170.       0.02114   0.    0.    0.
 180.       0.02427   0.    0.    0.
FB:
1    0.    0.    8.
2.4
ST:
1
1    0.5   110.   T    T    T    T
10.8      90.     3.65
0.    0.    8.
3
0.
10
  0.375     0.215
  0.125     0.215
 -0.125     0.215
 -0.375     0.215
 -0.375     0.000
 -0.375    -0.215
 -0.125    -0.215
  0.125    -0.215
  0.375    -0.215
  0.375     0.000
```

```
PZ:
1
90.
-180.   180.   0.5
F
PP:
2
1    1
1    2
4    1
1    2
XQ:
```

FRONT VIEW                SIDE VIEW
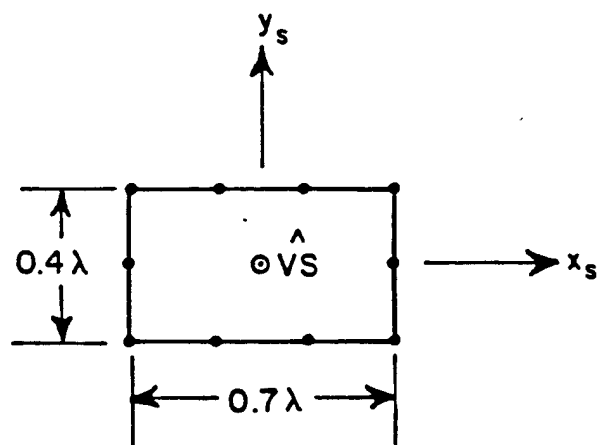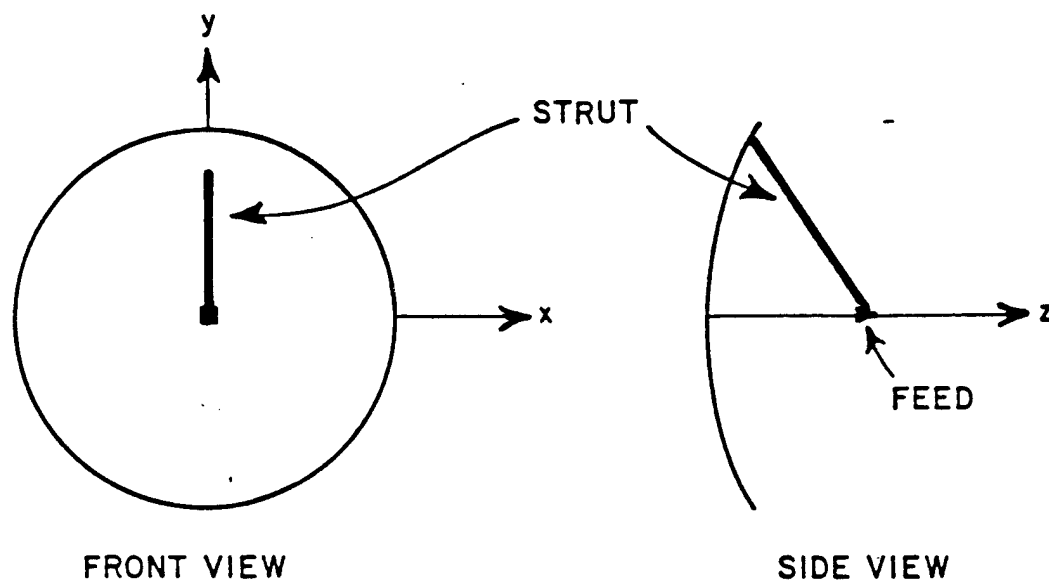
STRUT CROSS SECTION NM = 10

Figure 7.  Geometry of reflector with one rectangular strut.
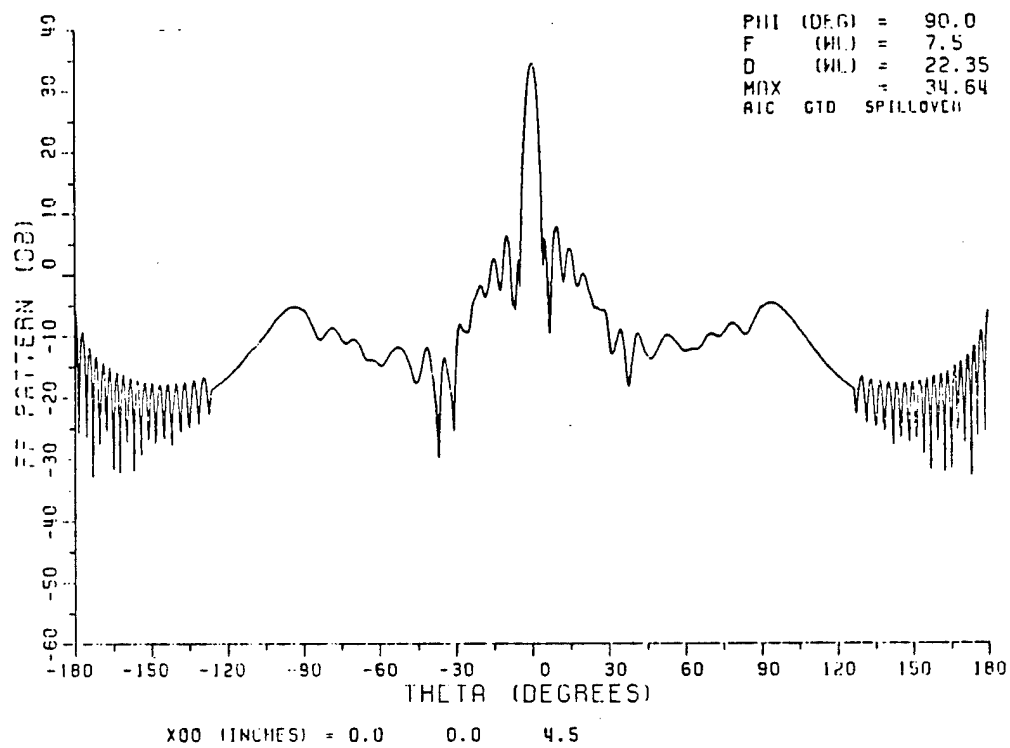
(a) total field
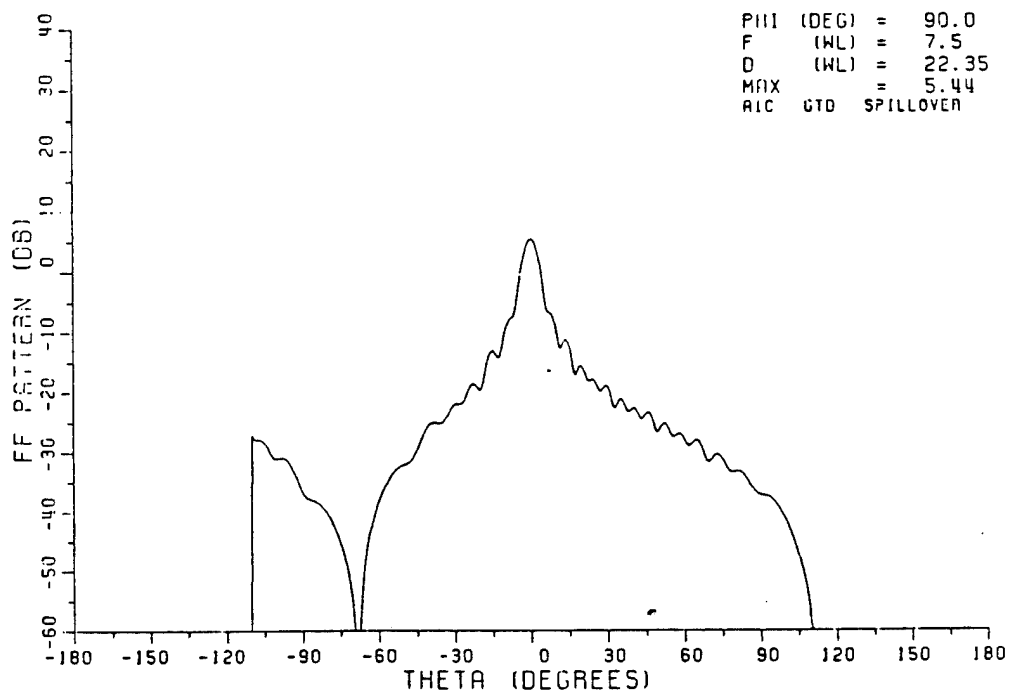


(b) strut scattering

Figure 8. Far field patterns of example 2 for PHI=90.0 degrees.

Example 2:

This example illustrates how the feed/strut scattering pattern (ISTRUT=2) of a reflector antenna can be calculated by using the ST: command. Struts with circular cross section (LGEN=F) are used in this example to model the hoop around the circumference of a 4-quadrant reflector (with a radius of 7.5 m) as shown in Figure 9. The feed is pointed toward the center of one quadrant. Consequently, the reflector rim corresponds to the single quadrant as shown in Figure 9. The surface which corresponds to the other three quadrants is open. The incident fields on the struts are from the feed directly, i.e., a feed/strut scattering case. The feed of this example is a 19 element array and the feed data are identical to the example in the AF: command. However, the increment in feed data for each cut is 1 degree in this example. The calculated patterns for the strut scattering, the reflector fields alone, and the total fields are shown in Figure 10. The input data for this example are given in Table 2.

## TABLE 2

## INPUT DATA FOR THE FEED/STRUT SCATTERING CALCULATION

```
CM:         ***** STRUT2.DAT *****
CM:     EXAMPLE OF FEED/STRUT SCATTERING
CM:        19 ELEMENTS ARRAY FEED
CE:
DG:
1
3    366.85   3.0    3.0    0.    22
 208.791        188.003
 191.766        203.741
 173.558        218.094
 154.281        230.975
 134.052        242.304
 112.997        252.011
  91.245        260.035
  68.931        266.329
  46.191        270.852
  23.167        273.577
   0.000        274.487
 -23.167        273.577
 -46.191        270.852
 -68.931        266.329
 -91.245        260.035
-112.997        252.011
-134.052        242.304
-154.281        230.975
-173.558        218.094
-191.766        203.741
-208.791        188.003
   0.000        -20.788
TO:
F   0.   0.
F   0    0    0    0
F   F    F    F    0
T   T    F    0.8
T   F    F    F    T    F    0.    0.
T   T    0.   0.
FD: FOCUSED 19 ELEMENTS ARRAY FEED
0    T
T    0    T    1    90.    0.    F
10   0. 10. 20. 30. 40. 50. 60. 70. 80. 90.
91
        0.000      12.514      90.000    -300.000      0.000
        1.000      12.476      90.000    -300.000      0.000
        2.000      12.360      90.000    -300.000      0.000
        3.000      12.166      90.000    -300.000      0.000
        4.000      11.894      90.000    -300.000      0.000
        5.000      11.541      90.000    -300.000      0.000
        6.000      11.107      90.000    -300.000      0.000
        7.000      10.590      90.000    -300.000      0.000
        8.000       9.988      90.000    -300.000      0.000
        9.000       9.299      90.000    -300.000      0.000
       10.000       8.517      90.000    -300.000      0.000
       11.000       7.640      90.000    -300.000      0.000
       12.000       6.662      90.000    -300.000      0.000
       13.000       5.579      90.000    -300.000      0.000
       14.000       4.385      90.000    -300.000      0.000
       15.000       3.071      90.000    -300.000      0.000
       16.000       1.629      90.000    -300.000      0.000
       17.000       0.049      90.000    -300.000      0.000
       18.000      -1.679      90.000    -300.000      0.000
       19.000      -3.567      90.000    -300.000      0.000
```

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 20.000 | -5.627 | 90.000 | -300.000 | 0.000 |
| 21.000 | -7.867 | 90.000 | -300.000 | 0.000 |
| 22.000 | -10.287 | 90.000 | -300.000 | 0.000 |
| 23.000 | -12.865 | 90.000 | -300.000 | 0.000 |
| 24.000 | -15.538 | 90.000 | -300.000 | 0.000 |
| 25.000 | -18.155 | 90.000 | -300.000 | 0.000 |
| 26.000 | -20.446 | 90.000 | -300.000 | 0.000 |
| 27.000 | -22.038 | 90.000 | -300.000 | 0.000 |
| 28.000 | -22.640 | 90.000 | -300.000 | 0.000 |
| 29.000 | -22.276 | 90.000 | -300.000 | 0.000 |
| 30.000 | -21.261 | 90.000 | -300.000 | 0.000 |
| 31.000 | -19.964 | 90.000 | -300.000 | 0.000 |
| 32.000 | -18.643 | 90.000 | -300.000 | 0.000 |
| 33.000 | -17.434 | 90.000 | -300.000 | 0.000 |
| 34.000 | -16.395 | 90.000 | -300.000 | 0.000 |
| 35.000 | -15.548 | 90.000 | -300.000 | 0.000 |
| 36.000 | -14.893 | 90.000 | -300.000 | 0.000 |
| 37.000 | -14.421 | 90.000 | -300.000 | 0.000 |
| 38.000 | -14.126 | 90.000 | -300.000 | 0.000 |
| 39.000 | -13.996 | 90.000 | -300.000 | 0.000 |
| 40.000 | -14.024 | 90.000 | -300.000 | 0.000 |
| 41.000 | -14.203 | 90.000 | -300.000 | 0.000 |
| 42.000 | -14.526 | 90.000 | -300.000 | 0.000 |
| 43.000 | -14.988 | 90.000 | -300.000 | 0.000 |
| 44.000 | -15.586 | 90.000 | -300.000 | 0.000 |
| 45.000 | -16.313 | 90.000 | -300.000 | 0.000 |
| 46.000 | -17.163 | 90.000 | -300.000 | 0.000 |
| 47.000 | -18.125 | 90.000 | -300.000 | 0.000 |
| 48.000 | -19.185 | 90.000 | -300.000 | 0.000 |
| 49.000 | -20.317 | 90.000 | -300.000 | 0.000 |
| 50.000 | -21.480 | 90.000 | -300.000 | 0.000 |
| 51.000 | -22.615 | 90.000 | -300.000 | 0.000 |
| 52.000 | -23.634 | 90.000 | -300.000 | 0.000 |
| 53.000 | -24.436 | 90.000 | -300.000 | 0.000 |
| 54.000 | -24.910 | 90.000 | -300.000 | 0.000 |
| 55.000 | -24.981 | 90.000 | -300.000 | 0.000 |
| 56.000 | -24.630 | 90.000 | -300.000 | 0.000 |
| 57.000 | -23.907 | 90.000 | -300.000 | 0.000 |
| 58.000 | -22.904 | 90.000 | -300.000 | 0.000 |
| 59.000 | -21.731 | 90.000 | -300.000 | 0.000 |
| 60.000 | -20.475 | 90.000 | -300.000 | 0.000 |
| 61.000 | -19.205 | 90.000 | -300.000 | 0.000 |
| 62.000 | -17.964 | 90.000 | -300.000 | 0.000 |
| 63.000 | -16.779 | 90.000 | -300.000 | 0.000 |
| 64.000 | -15.664 | 90.000 | -300.000 | 0.000 |
| 65.000 | -14.626 | 90.000 | -300.000 | 0.000 |
| 66.000 | -13.666 | 90.000 | -300.000 | 0.000 |
| 67.000 | -12.783 | 90.000 | -300.000 | 0.000 |
| 68.000 | -11.976 | 90.000 | -300.000 | 0.000 |
| 69.000 | -11.238 | 90.000 | -300.000 | 0.000 |
| 70.000 | -10.568 | 90.000 | -300.000 | 0.000 |
| 71.000 | -9.961 | 90.000 | -300.000 | 0.000 |
| 72.000 | -9.413 | 90.000 | -300.000 | 0.000 |
| 73.000 | -8.922 | 90.000 | -300.000 | 0.000 |
| 74.000 | -8.482 | 90.000 | -300.000 | 0.000 |
| 75.000 | -8.093 | 90.000 | -300.000 | 0.000 |
| 76.000 | -7.749 | 90.000 | -300.000 | 0.000 |
| 77.000 | -7.450 | 90.000 | -300.000 | 0.000 |
| 78.000 | -7.192 | 90.000 | -300.000 | 0.000 |
| 79.000 | -6.975 | 90.000 | -300.000 | 0.000 |
| 80.000 | -6.795 | 90.000 | -300.000 | 0.000 |
| 81.000 | -6.651 | 90.000 | -300.000 | 0.000 |
| 82.000 | -6.542 | 90.000 | -300.000 | 0.000 |
| 83.000 | -6.466 | 90.000 | -300.000 | 0.000 |
| 84.000 | -6.422 | 90.000 | -300.000 | 0.000 |
| 85.000 | -6.411 | 90.000 | -300.000 | 0.000 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 86.000 | -6.429 | 90.000 | -300.000 | 0.000 |
| 87.000 | -6.478 | 90.000 | -300.000 | ·0.000 |
| 88.000 | -6.557 | 90.000 | -300.000 | 0.000 |
| 89.000 | -6.665 | 90.000 | -300.000 | 0.000 |
| 90.000 | -292.618 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.514 | 90.000 | -71.604 | -89.544 |
| 1.000 | 12.476 | 90.000 | -76.165 | -87.245 |
| 2.000 | 12.360 | 90.000 | -69.907 | -50.021 |
| 3.000 | 12.166 | 90.000 | -68.169 | -129.977 |
| 4.000 | 11.893 | 90.000 | -67.223 | -129.970 |
| 5.000 | 11.540 | 90.000 | -66.675 | -129.970 |
| 6.000 | 11.106 | 90.000 | -66.398 | -129.970 |
| 7.000 | 10.588 | 90.000 | -66.295 | -129.972 |
| 8.000 | 9.986 | 90.000 | -66.405 | -129.975 |
| 9.000 | 9.295 | 90.000 | -66.646 | -129.975 |
| 10.000 | 8.511 | 90.000 | -67.046 | -129.976 |
| 11.000 | 7.632 | 90.000 | -67.579 | -129.977 |
| 12.000 | 6.651 | 90.000 | -68.244 | -129.979 |
| 13.000 | 5.563 | 90.000 | -69.046 | -129.980 |
| 14.000 | 4.361 | 90.000 | -69.993 | -129.981 |
| 15.000 | 3.036 | 90.000 | -71.082 | -129.981 |
| 16.000 | 1.576 | 90.000 | -72.327 | -129.982 |
| 17.000 | -0.032 | 90.000 | -73.736 | -129.982 |
| 18.000 | -1.804 | 90.000 | -75.325 | -129.983 |
| 19.000 | -3.762 | 90.000 | -77.114 | -129.983 |
| 20.000 | -5.932 | 90.000 | -79.130 | -129.984 |
| 21.000 | -8.352 | 90.000 | -81.406 | -129.984 |
| 22.000 | -11.069 | 90.000 | -83.991 | -129.984 |
| 23.000 | -14.147 | 90.000 | -86.947 | -129.985 |
| 24.000 | -17.674 | 90.000 | -90.361 | -129.986 |
| 25.000 | -21.768 | 90.000 | -94.351 | -129.986 |
| 26.000 | -26.556 | 90.000 | -99.043 | -129.986 |
| 27.000 | -32.040 | 90.000 | -104.439 | -129.987 |
| 28.000 | -37.378 | 90.000 | -109.697 | -129.987 |
| 29.000 | -39.553 | 90.000 | -111.798 | -129.987 |
| 30.000 | -37.055 | 90.000 | -109.235 | -129.987 |
| 31.000 | -33.082 | 90.000 | -105.202 | -129.987 |
| 32.000 | -29.566 | 90.000 | -101.630 | -129.987 |
| 33.000 | -26.798 | 90.000 | -98.814 | -129.988 |
| 34.000 | -24.685 | 90.000 | -96.659 | -129.988 |
| 35.000 | -23.106 | 90.000 | -95.042 | -129.988 |
| 36.000 | -21.964 | 90.000 | -93.868 | -129.988 |
| 37.000 | -21.193 | 90.000 | -93.069 | -129.989 |
| 38.000 | -20.747 | 90.000 | -92.600 | -129.989 |
| 39.000 | -20.598 | 90.000 | -92.434 | -129.989 |
| 40.000 | -20.735 | 90.000 | -92.556 | -129.989 |
| 41.000 | -21.158 | 90.000 | -92.969 | -129.989 |
| 42.000 | -21.885 | 90.000 | -93.690 | -129.989 |
| 43.000 | -22.956 | 90.000 | -94.759 | -129.989 |
| 44.000 | -24.447 | 90.000 | -96.252 | -129.989 |
| 45.000 | -26.506 | 90.000 | -98.316 | -129.990 |
| 46.000 | -29.448 | 90.000 | -101.266 | -129.990 |
| 47.000 | -34.127 | 90.000 | -105.958 | -129.990 |
| 48.000 | -44.964 | 90.000 | -116.809 | -129.990 |
| 49.000 | -42.040 | -90.000 | -113.902 | 50.009 |
| 50.000 | -33.721 | -90.000 | -105.604 | 50.009 |
| 51.000 | -29.870 | -90.000 | -101.776 | 50.009 |
| 52.000 | -27.513 | -90.000 | -99.445 | 50.009 |
| 53.000 | -25.952 | -90.000 | -97.912 | 50.009 |
| 54.000 | -24.912 | -90.000 | -96.903 | 50.009 |
| 55.000 | -24.262 | -90.000 | -96.285 | 50.009 |
| 56.000 | -23.932 | -90.000 | -95.989 | 50.009 |
| 57.000 | -23.887 | -90.000 | -95.980 | 50.009 |
| 58.000 | -24.116 | -90.000 | -96.246 | 50.009 |
| 59.000 | -24.629 | -90.000 | -96.799 | 50.008 |
| 60.000 | -25.461 | -90.000 | -97.671 | 50.008 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 61.000 | -26.683 | -90.000 | -98.934 | 50.008 |
| 62.000 | -28.435 | -90.000 | -100.729 | 50.008 |
| 63.000 | -31.020 | -90.000 | -103.357 | 50.008 |
| 64.000 | -35.244 | -90.000 | -107.625 | 50.008 |
| 65.000 | -45.077 | -90.000 | -117.502 | 50.008 |
| 66.000 | -43.314 | 90.000 | -115.784 | -129.992 |
| 67.000 | -34.103 | 90.000 | -106.617 | -129.992 |
| 68.000 | -29.655 | 90.000 | -102.213 | -129.993 |
| 69.000 | -26.705 | 90.000 | -99.307 | -129.992 |
| 70.000 | -24.514 | 90.000 | -97.159 | -129.993 |
| 71.000 | -22.791 | 90.000 | -95.480 | -129.993 |
| 72.000 | -21.388 | 90.000 | -94.119 | -129.993 |
| 73.000 | -20.223 | 90.000 | -92.995 | -129.993 |
| 74.000 | -19.242 | 90.000 | -92.053 | -129.993 |
| 75.000 | -18.410 | 90.000 | -91.260 | -129.993 |
| 76.000 | -17.700 | 90.000 | -90.587 | -129.993 |
| 77.000 | -17.096 | 90.000 | -90.018 | -129.993 |
| 78.000 | -16.582 | 90.000 | -89.538 | -129.993 |
| 79.000 | -16.149 | 90.000 | -89.136 | -129.993 |
| 80.000 | -15.787 | 90.000 | -88.803 | -129.993 |
| 81.000 | -15.489 | 90.000 | -88.531 | -129.993 |
| 82.000 | -15.251 | 90.000 | -88.317 | -129.994 |
| 83.000 | -15.067 | 90.000 | -88.155 | -129.994 |
| 84.000 | -14.934 | 90.000 | -88.041 | -129.994 |
| 85.000 | -14.849 | 90.000 | -87.972 | -129.994 |
| 86.000 | -14.810 | 90.000 | -87.946 | -129.994 |
| 87.000 | -14.816 | 90.000 | -87.962 | -129.994 |
| 88.000 | -14.864 | 90.000 | -88.018 | -129.994 |
| 89.000 | -14.953 | 90.000 | -88.112 | -129.994 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.514 | 90.000 | -74.179 | -89.097 |
| 1.000 | 12.476 | 90.000 | -83.262 | -84.500 |
| 2.000 | 12.359 | 90.000 | -70.625 | -10.052 |
| 3.000 | 12.165 | 90.000 | -66.949 | -169.963 |
| 4.000 | 11.892 | 90.000 | -64.777 | -169.950 |
| 5.000 | 11.539 | 90.000 | -63.315 | -169.950 |
| 6.000 | 11.104 | 90.000 | -62.314 | -169.951 |
| 7.000 | 10.586 | 90.000 | -61.576 | -169.954 |
| 8.000 | 9.982 | 90.000 | -61.175 | -169.958 |
| 9.000 | 9.290 | 90.000 | -60.946 | -169.960 |
| 10.000 | 8.504 | 90.000 | -60.940 | -169.962 |
| 11.000 | 7.621 | 90.000 | -61.104 | -169.965 |
| 12.000 | 6.635 | 90.000 | -61.429 | -169.967 |
| 13.000 | 5.538 | 90.000 | -61.923 | -169.969 |
| 14.000 | 4.322 | ·90.000 | -62.594 | -169.970 |
| 15.000 | 2.974 | 90.000 | -63.434 | -169.971 |
| 16.000 | 1.479 | 90.000 | -64.459 | -169.972 |
| 17.000 | -0.184 | 90.000 | -65.684 | -169.974 |
| 18.000 | -2.046 | 90.000 | -67.135 | -169.975 |
| 19.000 | -4.148 | 90.000 | -68.854 | -169.975 |
| 20.000 | -6.558 | 90.000 | -70.904 | -169.976 |
| 21.000 | -9.388 | 90.000 | -73.396 | -169.977 |
| 22.000 | -12.849 | 90.000 | -76.538 | -169.978 |
| 23.000 | -17.415 | 90.000 | -80.805 | -169.979 |
| 24.000 | -24.635 | 90.000 | -87.740 | -169.980 |
| 25.000 | -55.565 | -90.000 | -118.401 | 10.020 |
| 26.000 | -26.388 | -90.000 | -88.969 | 10.020 |
| 27.000 | -21.732 | -90.000 | -84.073 | 10.019 |
| 28.000 | -19.538 | -90.000 | -81.650 | 10.019 |
| 29.000 | -18.391 | -90.000 | -80.286 | 10.018 |
| 30.000 | -17.831 | -90.000 | -79.521 | 10.018 |
| 31.000 | -17.648 | -90.000 | -79.144 | 10.018 |
| 32.000 | -17.719 | -90.000 | -79.029 | 10.017 |
| 33.000 | -17.959 | -90.000 | -79.093 | 10.017 |
| 34.000 | -18.302 | -90.000 | -79.270 | 10.017 |
| 35.000 | -18.693 | -90.000 | -79.502 | 10.016 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 36.000 | -19.079 | -90.000 | -79.738 | 10.016 |
| 37.000 | -19.415 | -90.000 | -79.933 | 10.015 |
| 38.000 | -19.665 | -90.000 | -80.048 | 10.015 |
| 39.000 | -19.801 | -90.000 | -80.057 | 10.015 |
| 40.000 | -19.811 | -90.000 | -79.946 | 10.015 |
| 41.000 | -19.697 | -90.000 | -79.718 | 10.015 |
| 42.000 | -19.475 | -90.000 | -79.388 | 10.015 |
| 43.000 | -19.166 | -90.000 | -78.977 | 10.014 |
| 44.000 | -18.798 | -90.000 | -78.514 | 10.014 |
| 45.000 | -18.397 | -90.000 | -78.022 | 10.014 |
| 46.000 | -17.987 | -90.000 | -77.527 | 10.014 |
| 47.000 | -17.586 | -90.000 | -77.047 | 10.014 |
| 48.000 | -17.212 | -90.000 | -76.599 | 10.013 |
| 49.000 | -16.875 | -90.000 | -76.191 | 10.013 |
| 50.000 | -16.582 | -90.000 | -75.834 | 10.013 |
| 51.000 | -16.339 | -90.000 | -75.530 | 10.013 |
| 52.000 | -16.149 | -90.000 | -75.285 | 10.012 |
| 53.000 | -16.014 | -90.000 | -75.097 | 10.012 |
| 54.000 | -15.933 | -90.000 | -74.969 | 10.012 |
| 55.000 | -15.908 | -90.000 | -74.900 | 10.012 |
| 56.000 | -15.936 | -90.000 | -74.887 | 10.012 |
| 57.000 | -16.017 | -90.000 | -74.931 | 10.012 |
| 58.000 | -16.148 | -90.000 | -75.028 | 10.012 |
| 59.000 | -16.328 | -90.000 | -75.178 | 10.011 |
| 60.000 | -16.555 | -90.000 | -75.378 | 10.011 |
| 61.000 | -16.828 | -90.000 | -75.626 | 10.011 |
| 62.000 | -17.144 | -90.000 | -75.920 | 10.011 |
| 63.000 | -17.500 | -90.000 | -76.257 | 10.011 |
| 64.000 | -17.895 | -90.000 | -76.636 | 10.011 |
| 65.000 | -18.329 | -90.000 | -77.055 | 10.011 |
| 66.000 | -18.796 | -90.000 | -77.510 | 10.011 |
| 67.000 | -19.297 | -90.000 | -78.001 | 10.011 |
| 68.000 | -19.829 | -90.000 | -78.524 | 10.010 |
| 69.000 | -20.390 | -90.000 | -79.078 | 10.011 |
| 70.000 | -20.977 | -90.000 | -79.660 | 10.010 |
| 71.000 | -21.588 | -90.000 | -80.268 | 10.010 |
| 72.000 | -22.221 | -90.000 | -80.898 | 10.010 |
| 73.000 | -22.872 | -90.000 | -81.548 | 10.010 |
| 74.000 | -23.539 | -90.000 | -82.214 | 10.010 |
| 75.000 | -24.219 | -90.000 | -82.894 | 10.010 |
| 76.000 | -24.907 | -90.000 | -83.584 | 10.010 |
| 77.000 | -25.601 | -90.000 | -84.278 | 10.010 |
| 78.000 | -26.293 | -90.000 | -84.973 | 10.010 |
| 79.000 | -26.982 | -90.000 | -85.664 | 10.010 |
| 80.000 | -27.659 | -90.000 | -86.343 | 10.010 |
| 81.000 | -28.319 | -90.000 | -87.006 | 10.010 |
| 82.000 | -28.955 | -90.000 | -87.644 | 10.009 |
| 83.000 | -29.559 | -90.000 | -88.251 | 10.009 |
| 84.000 | -30.125 | -90.000 | -88.819 | 10.009 |
| 85.000 | -30.643 | -90.000 | -89.340 | 10.009 |
| 86.000 | -31.108 | -90.000 | -89.806 | 10.009 |
| 87.000 | -31.511 | -90.000 | -90.211 | 10.009 |
| 88.000 | -31.847 | -90.000 | -90.548 | 10.009 |
| 89.000 | -32.112 | -90.000 | -90.813 | 10.009 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.514 | 90.000 | -76.755 | -88.651 |
| 1.000 | 12.475 | 90.000 | -90.359 | -81.755 |
| 2.000 | 12.359 | 90.000 | -71.344 | 29.917 |
| 3.000 | 12.164 | 90.000 | -65.729 | 150.050 |
| 4.000 | 11.891 | 90.000 | -62.331 | 150.070 |
| 5.000 | 11.538 | 90.000 | -59.956 | 150.071 |
| 6.000 | 11.103 | 90.000 | -58.230 | 150.069 |
| 7.000 | 10.584 | 90.000 | -56.856 | 150.063 |
| 8.000 | 9.980 | 90.000 | -55.944 | 150.058 |
| 9.000 | 9.286 | 90.000 | -55.245 | 150.055 |
| 10.000 | 8.499 | 90.000 | -54.833 | 150.051 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 11.000 | 7.613 | 90.000 | -54.626 | 150.048 |
| 12.000 | 6.624 | 90.000 | -54.609 | 150.045 |
| 13.000 | 5.522 | 90.000 | -54.791 | 150.042 |
| 14.000 | 4.298 | 90.000 | -55.180 | 150.040 |
| 15.000 | 2.939 | 90.000 | -55.758 | 150.038 |
| 16.000 | 1.425 | 90.000 | -56.547 | 150.037 |
| 17.000 | -0.267 | 90.000 | -57.561 | 150.035 |
| 18.000 | -2.175 | 90.000 | -58.832 | 150.034 |
| 19.000 | -4.353 | 90.000 | -60.412 | 150.032 |
| 20.000 | -6.894 | 90.000 | -62.389 | 150.031 |
| 21.000 | -9.961 | 90.000 | -64.923 | 150.030 |
| 22.000 | -13.899 | 90.000 | -68.357 | 150.029 |
| 23.000 | -19.662 | 90.000 | -73.641 | 150.028 |
| 24.000 | -32.829 | 90.000 | -86.352 | 150.027 |
| 25.000 | -27.346 | -90.000 | -80.435 | -29.974 |
| 26.000 | -20.458 | -90.000 | -73.134 | -29.975 |
| 27.000 | -17.416 | -90.000 | -69.697 | -29.975 |
| 28.000 | -15.703 | -90.000 | -67.607 | -29.976 |
| 29.000 | -14.675 | -90.000 | -66.219 | -29.977 |
| 30.000 | -14.062 | -90.000 | -65.263 | -29.977 |
| 31.000 | -13.726 | -90.000 | -64.597 | -29.977 |
| 32.000 | -13.578 | -90.000 | -64.133 | -29.978 |
| 33.000 | -13.560 | -90.000 | -63.812 | -29.979 |
| 34.000 | -13.628 | -90.000 | -63.590 | -29.979 |
| 35.000 | -13.750 | -90.000 | -63.432 | -29.979 |
| 36.000 | -13.899 | -90.000 | -63.315 | -29.980 |
| 37.000 | -14.057 | -90.000 | -63.216 | -29.980 |
| 38.000 | -14.207 | -90.000 | -63.119 | -29.981 |
| 39.000 | -14.341 | -90.000 | -63.017 | -29.981 |
| 40.000 | -14.454 | -90.000 | -62.902 | -29.982 |
| 41.000 | -14.545 | -90.000 | -62.775 | -29.982 |
| 42.000 | -14.616 | -90.000 | -62.637 | -29.982 |
| 43.000 | -14.673 | -90.000 | -62.493 | -29.982 |
| 44.000 | -14.724 | -90.000 | -62.351 | -29.982 |
| 45.000 | -14.775 | -90.000 | -62.216 | -29.983 |
| 46.000 | -14.836 | -90.000 | -62.099 | -29.983 |
| 47.000 | -14.914 | -90.000 | -62.006 | -29.983 |
| 48.000 | -15.017 | -90.000 | -61.945 | -29.984 |
| 49.000 | -15.150 | -90.000 | -61.922 | -29.984 |
| 50.000 | -15.322 | -90.000 | -61.942 | -29.984 |
| 51.000 | -15.536 | -90.000 | -62.012 | -29.984 |
| 52.000 | -15.797 | -90.000 | -62.136 | -29.985 |
| 53.000 | -16.111 | -90.000 | -62.317 | -29.985 |
| 54.000 | -16.479 | -90.000 | -62.560 | -29.985 |
| 55.000 | -16.907 | -90.000 | -62.867 | -29.985 |
| 56.000 | -17.399 | -90.000 | -63.244 | -29.985 |
| 57.000 | -17.959 | -90.000 | -63.693 | -29.985 |
| 58.000 | -18.591 | -90.000 | -64.221 | -29.985 |
| 59.000 | -19.303 | -90.000 | -64.833 | -29.986 |
| 60.000 | -20.101 | -90.000 | -65.536 | -29.986 |
| 61.000 | -20.995 | -90.000 | -66.340 | -29.986 |
| 62.000 | -21.998 | -90.000 | -67.257 | -29.986 |
| 63.000 | -23.127 | -90.000 | -68.305 | -29.986 |
| 64.000 | -24.406 | -90.000 | -69.507 | -29.986 |
| 65.000 | -25.871 | -90.000 | -70.899 | -29.986 |
| 66.000 | -27.576 | -90.000 | -72.535 | -29.986 |
| 67.000 | -29.609 | -90.000 | -74.503 | -29.986 |
| 68.000 | -32.129 | -90.000 | -76.962 | -29.987 |
| 69.000 | -35.467 | -90.000 | -80.242 | -29.986 |
| 70.000 | -40.517 | -90.000 | -85.237 | -29.987 |
| 71.000 | -52.100 | -90.000 | -96.771 | -29.988 |
| 72.000 | -48.064 | 90.000 | -92.687 | 150.012 |
| 73.000 | -40.109 | 90.000 | -84.688 | 150.013 |
| 74.000 | -36.314 | 90.000 | -80.853 | 150.013 |
| 75.000 | -33.898 | 90.000 | -78.399 | 150.013 |
| 76.000 | -32.184 | 90.000 | -76.650 | 150.013 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 77.000 | -30.901 | 90.000 | -75.334 | 150.013 |
| 78.000 | -29.908 | 90.000 | -74.313 | 150.013 |
| 79.000 | -29.131 | 90.000 | -73.508 | 150.013 |
| 80.000 | -28.515 | 90.000 | -72.869 | 150.013 |
| 81.000 | -28.030 | 90.000 | -72.363 | 150.013 |
| 82.000 | -27.652 | 90.000 | -71.965 | 150.012 |
| 83.000 | -27.364 | 90.000 | -71.661 | 150.012 |
| 84.000 | -27.154 | 90.000 | -71.436 | 150.012 |
| 85.000 | -27.012 | 90.000 | -71.282 | 150.012 |
| 86.000 | -26.932 | 90.000 | -71.192 | 150.012 |
| 87.000 | -26.908 | 90.000 | -71.160 | 150.012 |
| 88.000 | -26.936 | 90.000 | -71.184 | 150.012 |
| 89.000 | -27.015 | 90.000 | -71.259 | 150.012 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.513 | 90.000 | -79.330 | -88.204 |
| 1.000 | 12.475 | 90.000 | -97.456 | -79.010 |
| 2.000 | 12.359 | 90.000 | -72.063 | 69.886 |
| 3.000 | 12.164 | 90.000 | -64.510 | 110.063 |
| 4.000 | 11.890 | 90.000 | -59.885 | 110.090 |
| 5.000 | 11.536 | 90.000 | -56.597 | 110.091 |
| 6.000 | 11.101 | 90.000 | -54.147 | 110.088 |
| 7.000 | 10.582 | 90.000 | -52.136 | 110.081 |
| 8.000 | 9.978 | 90.000 | -50.713 | 110.075 |
| 9.000 | 9.283 | 90.000 | -49.542 | 110.070 |
| 10.000 | 8.497 | 90.000 | -48.723 | 110.065 |
| 11.000 | 7.611 | 90.000 | -48.142 | 110.061 |
| 12.000 | 6.624 | 90.000 | -47.778 | 110.057 |
| 13.000 | 5.525 | 90.000 | -47.640 | 110.053 |
| 14.000 | 4.307 | 90.000 | -47.733 | 110.051 |
| 15.000 | 2.958 | 90.000 | -48.029 | 110.048 |
| 16.000 | 1.460 | 90.000 | -48.548 | 110.046 |
| 17.000 | -0.205 | 90.000 | -49.294 | 110.044 |
| 18.000 | -2.069 | 90.000 | -50.294 | 110.042 |
| 19.000 | -4.173 | 90.000 | -51.585 | 110.040 |
| 20.000 | -6.585 | 90.000 | -53.229 | 110.038 |
| 21.000 | -9.418 | 90.000 | -55.334 | 110.037 |
| 22.000 | -12.880 | 90.000 | -58.105 | 110.036 |
| 23.000 | -17.447 | 90.000 | -62.014 | 110.035 |
| 24.000 | -24.660 | 90.000 | -68.600 | 110.033 |
| 25.000 | -56.231 | -90.000 | -99.573 | -69.967 |
| 26.000 | -26.457 | -90.000 | -69.227 | -69.969 |
| 27.000 | -21.796 | -90.000 | -64.018 | -69.970 |
| 28.000 | -19.603 | -90.000 | -61.300 | -69.971 |
| 29.000 | -18.457 | -90.000 | -59.651 | -69.972 |
| 30.000 | -17.900 | -90.000 | -58.612 | -69.972 |
| 31.000 | -17.720 | -90.000 | -57.967 | -69.972 |
| 32.000 | -17.794 | -90.000 | -57.593 | -69.973 |
| 33.000 | -18.037 | -90.000 | -57.407 | -69.974 |
| 34.000 | -18.384 | -90.000 | -57.339 | -69.974 |
| 35.000 | -18.778 | -90.000 | -57.333 | -69.975 |
| 36.000 | -19.167 | -90.000 | -57.338 | -69.976 |
| 37.000 | -19.507 | -90.000 | -57.307 | -69.976 |
| 38.000 | -19.760 | -90.000 | -57.201 | -69.977 |
| 39.000 | -19.899 | -90.000 | -56.996 | -69.977 |
| 40.000 | -19.913 | -90.000 | -56.675 | -69.978 |
| 41.000 | -19.803 | -90.000 | -56.243 | -69.978 |
| 42.000 | -19.584 | -90.000 | -55.713 | -69.978 |
| 43.000 | -19.280 | -90.000 | -55.108 | -69.979 |
| 44.000 | -18.917 | -90.000 | -54.454 | -69.979 |
| 45.000 | -18.520 | -90.000 | -53.776 | -69.980 |
| 46.000 | -18.115 | -90.000 | -53.100 | -69.980 |
| 47.000 | -17.719 | -90.000 | -52.442 | -69.980 |
| 48.000 | -17.350 | -90.000 | -51.820 | -69.981 |
| 49.000 | -17.017 | -90.000 | -51.243 | -69.981 |
| 50.000 | -16.730 | -90.000 | -50.719 | -69.981 |
| 51.000 | -16.491 | -90.000 | -50.254 | -69.981 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 52.000 | -16.307 | -90.000 | -49.849 | -69.982 |
| 53.000 | -16.177 | -90.000 | -49.507 | -69.982 |
| 54.000 | -16.101 | -90.000 | -49.227 | -69.982 |
| 55.000 | -16.081 | -90.000 | -49.009 | -69.982 |
| 56.000 | -16.113 | -90.000 | -48.852 | -69.982 |
| 57.000 | -16.198 | -90.000 | -48.754 | -69.982 |
| 58.000 | -16.334 | -90.000 | -48.713 | -69.982 |
| 59.000 | -16.519 | -90.000 | -48.729 | -69.983 |
| 60.000 | -16.750 | -90.000 | -48.798 | -69.983 |
| 61.000 | -17.027 | -90.000 | -48.919 | -69.983 |
| 62.000 | -17.347 | -90.000 | -49.088 | -69.983 |
| 63.000 | -17.707 | -90.000 | -49.305 | -69.983 |
| 64.000 | -18.106 | -90.000 | -49.567 | -69.983 |
| 65.000 | -18.543 | -90.000 | -49.872 | -69.983 |
| 66.000 | -19.015 | -90.000 | -50.218 | -69.983 |
| 67.000 | -19.519 | -90.000 | -50.603 | -69.983 |
| 68.000 | -20.054 | -90.000 | -51.024 | -69.984 |
| 69.000 | -20.618 | -90.000 | -51.479 | -69.983 |
| 70.000 | -21.208 | -90.000 | -51.966 | -69.984 |
| 71.000 | -21.822 | -90.000 | -52.484 | -69.984 |
| 72.000 | -22.458 | -90.000 | -53.027 | -69.984 |
| 73.000 | -23.111 | -90.000 | -53.594 | -69.984 |
| 74.000 | -23.780 | -90.000 | -54.183 | -69.984 |
| 75.000 | -24.462 | -90.000 | -54.788 | -69.984 |
| 76.000 | -25.152 | -90.000 | -55.408 | -69.984 |
| 77.000 | -25.847 | -90.000 | -56.036 | -69.984 |
| 78.000 | -26.541 | -90.000 | -56.671 | -69.984 |
| 79.000 | -27.231 | -90.000 | -57.304 | -69.984 |
| 80.000 | -27.908 | -90.000 | -57.932 | -69.984 |
| 81.000 | -28.569 | -90.000 | -58.547 | -69.984 |
| 82.000 | -29.206 | -90.000 | -59.142 | -69.985 |
| 83.000 | -29.811 | -90.000 | -59.712 | -69.985 |
| 84.000 | -30.376 | -90.000 | -60.246 | -69.985 |
| 85.000 | -30.895 | -90.000 | -60.739 | -69.985 |
| 86.000 | -31.359 | -90.000 | -61.181 | -69.985 |
| 87.000 | -31.763 | -90.000 | -61.568 | -69.985 |
| 88.000 | -32.099 | -90.000 | -61.892 | -69.985 |
| 89.000 | -32.363 | -90.000 | -62.149 | -69.985 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.513 | 90.000 | -79.643 | -88.204 |
| 1.000 | 12.475 | 90.000 | -97.769 | -79.057 |
| 2.000 | 12.359 | 90.000 | -72.376 | 109.792 |
| 3.000 | 12.163 | 90.000 | -64.822 | 109.923 |
| 4.000 | 11.889 | 90.000 | -60.198 | 109.903 |
| 5.000 | 11.535 | 90.000 | -56.909 | 109.857 |
| 6.000 | 11.100 | 90.000 | -54.459 | 109.807 |
| 7.000 | 10.580 | 90.000 | -52.447 | 109.752 |
| 8.000 | 9.976 | 90.000 | -51.024 | 109.699 |
| 9.000 | 9.281 | 90.000 | -49.852 | 109.646 |
| 10.000 | 8.496 | 90.000 | -49.030 | 109.594 |
| 11.000 | 7.613 | 90.000 | -48.446 | 109.541 |
| 12.000 | 6.629 | 90.000 | -48.077 | 109.490 |
| 13.000 | 5.537 | 90.000 | -47.930 | 109.438 |
| 14.000 | 4.332 | 90.000 | -48.009 | 109.386 |
| 15.000 | 3.003 | 90.000 | -48.283 | 109.335 |
| 16.000 | 1.538 | 90.000 | -48.767 | 109.284 |
| 17.000 | -0.074 | 90.000 | -49.458 | 109.232 |
| 18.000 | -1.851 | 90.000 | -50.368 | 109.180 |
| 19.000 | -3.813 | 90.000 | -51.516 | 109.129 |
| 20.000 | -5.989 | 90.000 | -52.920 | 109.077 |
| 21.000 | -8.414 | 90.000 | -54.615 | 109.025 |
| 22.000 | -11.135 | 90.000 | -56.643 | 108.972 |
| 23.000 | -14.216 | 90.000 | -59.063 | 108.920 |
| 24.000 | -17.745 | 90.000 | -61.962 | 108.866 |
| 25.000 | -21.838 | 90.000 | -65.453 | 108.813 |
| 26.000 | -26.619 | 90.000 | -69.659 | 108.759 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 27.000 | −32.083 | 90.000 | −74.573 | 108.705 |
| 28.000 | −37.386 | 90.000 | −79.347 | 108.651 |
| 29.000 | −39.553 | 90.000 | −81.007 | 108.596 |
| 30.000 | −37.106 | 90.000 | −78.074 | 108.540 |
| 31.000 | −33.178 | 90.000 | −73.678 | 108.485 |
| 32.000 | −29.690 | 90.000 | −69.738 | 108.429 |
| 33.000 | −26.941 | 90.000 | −66.555 | 108.372 |
| 34.000 | −24.843 | 90.000 | −64.038 | 108.316 |
| 35.000 | −23.276 | 90.000 | −62.068 | 108.258 |
| 36.000 | −22.146 | 90.000 | −60.548 | 108.200 |
| 37.000 | −21.386 | 90.000 | −59.413 | 108.142 |
| 38.000 | −20.952 | 90.000 | −58.615 | 108.083 |
| 39.000 | −20.814 | 90.000 | −58.128 | 108.024 |
| 40.000 | −20.963 | 90.000 | −57.937 | 107.964 |
| 41.000 | −21.397 | 90.000 | −58.044 | 107.904 |
| 42.000 | −22.137 | 90.000 | −58.467 | 107.844 |
| 43.000 | −23.221 | 90.000 | −59.245 | 107.783 |
| 44.000 | −24.729 | 90.000 | −60.456 | 107.722 |
| 45.000 | −26.807 | 90.000 | −62.247 | 107.660 |
| 46.000 | −29.777 | 90.000 | −64.941 | 107.598 |
| 47.000 | −34.511 | 90.000 | −69.407 | 107.537 |
| 48.000 | −45.627 | 90.000 | −80.264 | 107.474 |
| 49.000 | −42.077 | −90.000 | −76.463 | −72.589 |
| 50.000 | −33.927 | −90.000 | −68.070 | −72.651 |
| 51.000 | −30.121 | −90.000 | −64.031 | −72.714 |
| 52.000 | −27.789 | −90.000 | −61.472 | −72.777 |
| 53.000 | −26.246 | −90.000 | −59.710 | −72.840 |
| 54.000 | −25.220 | −90.000 | −58.474 | −72.902 |
| 55.000 | −24.582 | −90.000 | −57.631 | −72.965 |
| 56.000 | −24.261 | −90.000 | −57.113 | −73.027 |
| 57.000 | −24.224 | −90.000 | −56.886 | −73.090 |
| 58.000 | −24.460 | −90.000 | −56.939 | −73.151 |
| 59.000 | −24.979 | −90.000 | −57.281 | −73.213 |
| 60.000 | −25.813 | −90.000 | −57.947 | −73.273 |
| 61.000 | −27.034 | −90.000 | −59.004 | −73.334 |
| 62.000 | −28.779 | −90.000 | −60.592 | −73.394 |
| 63.000 | −31.342 | −90.000 | −63.004 | −73.453 |
| 64.000 | −35.501 | −90.000 | −67.020 | −73.510 |
| 65.000 | −44.958 | −90.000 | −76.336 | −73.567 |
| 66.000 | −44.274 | 90.000 | −75.521 | 106.377 |
| 67.000 | −34.742 | 90.000 | −65.862 | 106.323 |
| 68.000 | −30.232 | 90.000 | −61.231 | 106.269 |
| 69.000 | −27.261 | 90.000 | −58.144 | 106.218 |
| 70.000 | −25.060 | 90.000 | −55.834 | 106.167 |
| 71.000 | −23.333 | 90.000 | −54.004 | 106.118 |
| 72.000 | −21.931 | 90.000 | −52.502 | 106.071 |
| 73.000 | −20.766 | 90.000 | −51.246 | 106.026 |
| 74.000 | −19.787 | 90.000 | −50.180 | 105.983 |
| 75.000 | −18.958 | 90.000 | −49.268 | 105.942 |
| 76.000 | −18.251 | 90.000 | −48.486 | 105.903 |
| 77.000 | −17.650 | 90.000 | −47.814 | 105.866 |
| 78.000 | −17.139 | 90.000 | −47.238 | 105.832 |
| 79.000 | −16.709 | 90.000 | −46.747 | 105.800 |
| 80.000 | −16.349 | 90.000 | −46.333 | 105.770 |
| 81.000 | −16.055 | 90.000 | −45.989 | 105.743 |
| 82.000 | −15.819 | 90.000 | −45.708 | 105.719 |
| 83.000 | −15.637 | 90.000 | −45.488 | 105.697 |
| 84.000 | −15.506 | 90.000 | −45.323 | 105.678 |
| 85.000 | −15.423 | 90.000 | −45.212 | 105.662 |
| 86.000 | −15.385 | 90.000 | −45.150 | 105.649 |
| 87.000 | −15.392 | 90.000 | −45.139 | 105.639 |
| 88.000 | −15.440 | 90.000 | −45.174 | 105.632 |
| 89.000 | −15.531 | 90.000 | −45.256 | 105.627 |
| 90.000 | −300.000 | 0.000 | −300.000 | 0.000 |
| 0.000 | 12.513 | 90.000 | −77.694 | −88.649 |
| 1.000 | 12.474 | 90.000 | −91.298 | −81.895 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 2.000 | 12.358 | 90.000 | -72.283 | 149.637 |
| 3.000 | 12.162 | 90.000 | -66.668 | 149.629 |
| 4.000 | 11.888 | 90.000 | -63.270 | 149.508 |
| 5.000 | 11.534 | 90.000 | -60.893 | 149.368 |
| 6.000 | 11.098 | 90.000 | -59.167 | 149.224 |
| 7.000 | 10.578 | 90.000 | -57.791 | 149.077 |
| 8.000 | 9.973 | 90.000 | -56.877 | 148.930 |
| 9.000 | 9.279 | 90.000 | -56.175 | 148.783 |
| 10.000 | 8.493 | 90.000 | -55.759 | 148.637 |
| 11.000 | 7.611 | 90.000 | -55.544 | 148.490 |
| 12.000 | 6.629 | 90.000 | -55.516 | 148.343 |
| 13.000 | 5.540 | 90.000 | -55.680 | 148.195 |
| 14.000 | 4.340 | 90.000 | -56.040 | 148.047 |
| 15.000 | 3.020 | 90.000 | -56.574 | 147.899 |
| 16.000 | 1.571 | 90.000 | -57.292 | 147.750 |
| 17.000 | -0.015 | 90.000 | -58.194 | 147.600 |
| 18.000 | -1.750 | 90.000 | -59.285 | 147.449 |
| 19.000 | -3.645 | 90.000 | -60.576 | 147.298 |
| 20.000 | -5.713 | 90.000 | -62.072 | 147.146 |
| 21.000 | -7.961 | 90.000 | -63.779 | 146.993 |
| 22.000 | -10.388 | 90.000 | -65.694 | 146.838 |
| 23.000 | -12.973 | 90.000 | -67.791 | 146.683 |
| 24.000 | -15.652 | 90.000 | -70.005 | 146.526 |
| 25.000 | -18.275 | 90.000 | -72.185 | 146.368 |
| 26.000 | -20.572 | 90.000 | -74.059 | 146.210 |
| 27.000 | -22.171 | 90.000 | -75.255 | 146.050 |
| 28.000 | -22.785 | 90.000 | -75.481 | 145.888 |
| 29.000 | -22.434 | 90.000 | -74.760 | 145.725 |
| 30.000 | -21.436 | 90.000 | -73.407 | 145.561 |
| 31.000 | -20.157 | 90.000 | -71.786 | 145.396 |
| 32.000 | -18.851 | 90.000 | -70.152 | 145.229 |
| 33.000 | -17.658 | 90.000 | -68.643 | 145.061 |
| 34.000 | -16.635 | 90.000 | -67.317 | 144.891 |
| 35.000 | -15.803 | 90.000 | -66.193 | 144.720 |
| 36.000 | -15.161 | 90.000 | -65.271 | 144.547 |
| 37.000 | -14.706 | 90.000 | -64.544 | 144.374 |
| 38.000 | -14.425 | 90.000 | -64.002 | 144.198 |
| 39.000 | -14.310 | 90.000 | -63.637 | 144.022 |
| 40.000 | -14.353 | 90.000 | -63.437 | 143.844 |
| 41.000 | -14.547 | 90.000 | -63.397 | 143.665 |
| 42.000 | -14.886 | 90.000 | -63.510 | 143.485 |
| 43.000 | -15.364 | 90.000 | -63.771 | 143.303 |
| 44.000 | -15.978 | 90.000 | -64.174 | 143.120 |
| 45.000 | -16.721 | 90.000 | -64.715 | 142.936 |
| 46.000 | -17.587 | 90.000 | -65.386 | 142.751 |
| 47.000 | -18.567 | 90.000 | -66.177 | 142.566 |
| 48.000 | -19.646 | 90.000 | -67.073 | 142.379 |
| 49.000 | -20.796 | 90.000 | -68.048 | 142.192 |
| 50.000 | -21.979 | 90.000 | -69.061 | 142.005 |
| 51.000 | -23.133 | 90.000 | -70.052 | 141.817 |
| 52.000 | -24.174 | 90.000 | -70.936 | 141.629 |
| 53.000 | -24.996 | 90.000 | -71.606 | 141.441 |
| 54.000 | -25.491 | 90.000 | -71.955 | 141.253 |
| 55.000 | -25.580 | 90.000 | -71.902 | 141.066 |
| 56.000 | -25.244 | 90.000 | -71.430 | 140.879 |
| 57.000 | -24.533 | 90.000 | -70.588 | 140.692 |
| 58.000 | -23.542 | 90.000 | -69.471 | 140.508 |
| 59.000 | -22.378 | 90.000 | -68.186 | 140.324 |
| 60.000 | -21.132 | 90.000 | -66.824 | 140.142 |
| 61.000 | -19.871 | 90.000 | -65.451 | 139.961 |
| 62.000 | -18.640 | 90.000 | -64.113 | 139.782 |
| 63.000 | -17.464 | 90.000 | -62.835 | 139.606 |
| 64.000 | -16.359 | 90.000 | -61.631 | 139.433 |
| 65.000 | -15.331 | 90.000 | -60.508 | 139.263 |
| 66.000 | -14.381 | 90.000 | -59.469 | 139.095 |
| 67.000 | -13.509 | 90.000 | -58.510 | 138.932 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 68.000 | -12.711 | 90.000 | -57.630 | 138.772 |
| 69.000 | -11.983 | 90.000 | -56.825 | 138.617 |
| 70.000 | -11.322 | 90.000 | -56.090 | 138.466 |
| 71.000 | -10.725 | 90.000 | -55.423 | 138.319 |
| 72.000 | -10.187 | 90.000 | -54.818 | 138.179 |
| 73.000 | -9.703 | 90.000 | -54.273 | 138.044 |
| 74.000 | -9.272 | 90.000 | -53.783 | 137.914 |
| 75.000 | -8.891 | 90.000 | -53.346 | 137.791 |
| 76.000 | -8.554 | 90.000 | -52.959 | 137.674 |
| 77.000 | -8.262 | 90.000 | -52.619 | 137.564 |
| 78.000 | -8.011 | 90.000 | -52.324 | 137.461 |
| 79.000 | -7.800 | 90.000 | -52.072 | 137.365 |
| 80.000 | -7.625 | 90.000 | -51.861 | 137.276 |
| 81.000 | -7.487 | 90.000 | -51.689 | 137.195 |
| 82.000 | -7.382 | 90.000 | -51.554 | 137.122 |
| 83.000 | -7.310 | 90.000 | -51.457 | 137.058 |
| 84.000 | -7.270 | 90.000 | -51.394 | 137.001 |
| 85.000 | -7.262 | 90.000 | -51.366 | 136.953 |
| 86.000 | -7.283 | 90.000 | -51.371 | 136.914 |
| 87.000 | -7.334 | 90.000 | -51.410 | 136.883 |
| 88.000 | -7.414 | 90.000 | -51.481 | 136.861 |
| 89.000 | -7.523 | 90.000 | -51.585 | 136.847 |
| 90.000 | -292.626 | 0.000 | -292.626 | 0.000 |
| 0.000 | 12.512 | 90.000 | -75.745 | -89.095 |
| 1.000 | 12.474 | 90.000 | -84.827 | -84.732 |
| 2.000 | 12.357 | 90.000 | -72.190 | -170.519 |
| 3.000 | 12.162 | 90.000 | -68.514 | -170.665 |
| 4.000 | 11.887 | 90.000 | -66.341 | -170.887 |
| 5.000 | 11.533 | 90.000 | -64.878 | -171.121 |
| 6.000 | 11.096 | 90.000 | -63.875 | -171.359 |
| 7.000 | 10.576 | 90.000 | -63.135 | -171.598 |
| 8.000 | 9.970 | 90.000 | -62.732 | -171.839 |
| 9.000 | 9.274 | 90.000 | -62.500 | -172.079 |
| 10.000 | 8.487 | 90.000 | -62.490 | -172.320 |
| 11.000 | 7.603 | 90.000 | -62.649 | -172.561 |
| 12.000 | 6.618 | 90.000 | -62.966 | -172.804 |
| 13.000 | 5.524 | 90.000 | -63.449 | -173.048 |
| 14.000 | 4.316 | 90.000 | -64.103 | -173.292 |
| 15.000 | 2.985 | 90.000 | -64.917 | -173.537 |
| 16.000 | 1.518 | 90.000 | -65.904 | -173.783 |
| 17.000 | -0.096 | 90.000 | -67.070 | -174.032 |
| 18.000 | -1.875 | 90.000 | -68.428 | -174.282 |
| 19.000 | -3.840 | 90.000 | -69.998 | -174.533 |
| 20.000 | -6.018 | 90.000 | -71.804 | -174.785 |
| 21.000 | -8.446 | 90.000 | -73.881 | -175.040 |
| 22.000 | -11.169 | 90.000 | -76.272 | -175.296 |
| 23.000 | -14.252 | 90.000 | -79.040 | -175.554 |
| 24.000 | -17.783 | 90.000 | -82.272 | -175.814 |
| 25.000 | -21.875 | 90.000 | -86.080 | -176.076 |
| 26.000 | -26.652 | 90.000 | -90.587 | -176.340 |
| 27.000 | -32.102 | 90.000 | -95.779 | -176.606 |
| 28.000 | -37.373 | 90.000 | -100.804 | -176.875 |
| 29.000 | -39.517 | 90.000 | -102.713 | -177.145 |
| 30.000 | -37.091 | 90.000 | -100.065 | -177.418 |
| 31.000 | -33.192 | 90.000 | -95.951 | -177.693 |
| 32.000 | -29.723 | 90.000 | -92.277 | -177.970 |
| 33.000 | -26.987 | 90.000 | -89.344 | -178.251 |
| 34.000 | -24.899 | 90.000 | -87.068 | -178.533 |
| 35.000 | -23.339 | 90.000 | -85.328 | -178.818 |
| 36.000 | -22.215 | 90.000 | -84.032 | -179.105 |
| 37.000 | -21.461 | 90.000 | -83.112 | -179.394 |
| 38.000 | -21.031 | 90.000 | -82.523 | -179.686 |
| 39.000 | -20.898 | 90.000 | -82.238 | -179.979 |
| 40.000 | -21.050 | 90.000 | -82.243 | 179.724 |
| 41.000 | -21.488 | 90.000 | -82.541 | 179.426 |
| 42.000 | -22.229 | 90.000 | -83.148 | 179.125 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 43.000 | -23.314 | 90.000 | -84.104 | 178.822 |
| 44.000 | -24.819 | 90.000 | -85.485 | 178.519 |
| 45.000 | -26.891 | 90.000 | -87.438 | 178.212 |
| 46.000 | -29.845 | 90.000 | -90.278 | 177.904 |
| 47.000 | -34.531 | 90.000 | -94.855 | 177.595 |
| 48.000 | -45.337 | 90.000 | -105.555 | 177.285 |
| 49.000 | -42.538 | -90.000 | -102.656 | -3.027 |
| 50.000 | -34.205 | -90.000 | -94.226 | -3.339 |
| 51.000 | -30.362 | -90.000 | -90.291 | -3.652 |
| 52.000 | -28.017 | -90.000 | -87.856 | -3.965 |
| 53.000 | -26.468 | -90.000 | -86.222 | -4.278 |
| 54.000 | -25.441 | -90.000 | -85.114 | -4.591 |
| 55.000 | -24.804 | -90.000 | -84.399 | -4.903 |
| 56.000 | -24.486 | -90.000 | -84.006 | -5.214 |
| 57.000 | -24.453 | -90.000 | -83.900 | -5.525 |
| 58.000 | -24.693 | -90.000 | -84.072 | -5.833 |
| 59.000 | -25.217 | -90.000 | -84.530 | -6.140 |
| 60.000 | -26.058 | -90.000 | -85.308 | -6.443 |
| 61.000 | -27.286 | -90.000 | -86.477 | -6.744 |
| 62.000 | -29.041 | -90.000 | -88.175 | -7.042 |
| 63.000 | -31.620 | -90.000 | -90.698 | -7.335 |
| 64.000 | -35.811 | -90.000 | -94.838 | -7.624 |
| 65.000 | -45.415 | -90.000 | -104.390 | -7.908 |
| 66.000 | -44.327 | 90.000 | -103.256 | 171.814 |
| 67.000 | -34.927 | 90.000 | -93.811 | 171.541 |
| 68.000 | -30.448 | 90.000 | -89.289 | 171.275 |
| 69.000 | -27.491 | 90.000 | -86.291 | 171.016 |
| 70.000 | -25.300 | 90.000 | -84.062 | 170.765 |
| 71.000 | -23.581 | 90.000 | -82.306 | 170.521 |
| 72.000 | -22.184 | 90.000 | -80.875 | 170.286 |
| 73.000 | -21.025 | 90.000 | -79.684 | 170.061 |
| 74.000 | -20.051 | 90.000 | -78.680 | 169.845 |
| 75.000 | -19.225 | 90.000 | -77.825 | 169.640 |
| 76.000 | -18.522 | 90.000 | -77.097 | 169.445 |
| 77.000 | -17.925 | 90.000 | -76.474 | 169.261 |
| 78.000 | -17.417 | 90.000 | -75.944 | 169.090 |
| 79.000 | -16.989 | 90.000 | -75.495 | 168.930 |
| 80.000 | -16.632 | 90.000 | -75.120 | 168.782 |
| 81.000 | -16.340 | 90.000 | -74.810 | 168.648 |
| 82.000 | -16.106 | 90.000 | -74.561 | 168.526 |
| 83.000 | -15.926 | 90.000 | -74.367 | 168.418 |
| 84.000 | -15.796 | 90.000 | -74.226 | 168.324 |
| 85.000 | -15.714 | 90.000 | -74.135 | 168.244 |
| 86.000 | -15.678 | 90.000 | -74.090 | 168.178 |
| 87.000 | -15.685 | 90.000 | -74.091 | 168.126 |
| 88.000 | -15.734 | 90.000 | -74.136 | 168.090 |
| 89.000 | -15.825 | 90.000 | -74.224 | 168.068 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.512 | 90.000 | -73.796 | -89.540 |
| 1.000 | 12.473 | 90.000 | -78.357 | -87.570 |
| 2.000 | 12.357 | 90.000 | -72.098 | -130.674 |
| 3.000 | 12.161 | 90.000 | -70.359 | -130.959 |
| 4.000 | 11.886 | 90.000 | -69.413 | -131.281 |
| 5.000 | 11.531 | 90.000 | -68.862 | -131.610 |
| 6.000 | 11.094 | 90.000 | -68.583 | -131.941 |
| 7.000 | 10.573 | 90.000 | -68.478 | -132.274 |
| 8.000 | 9.966 | 90.000 | -68.586 | -132.607 |
| 9.000 | 9.269 | 90.000 | -68.826 | -132.942 |
| 10.000 | 8.479 | 90.000 | -69.224 | -133.277 |
| 11.000 | 7.592 | 90.000 | -69.756 | -133.613 |
| 12.000 | 6.601 | 90.000 | -70.421 | -133.951 |
| 13.000 | 5.498 | 90.000 | -71.227 | -134.290 |
| 14.000 | 4.276 | 90.000 | -72.182 | -134.631 |
| 15.000 | 2.922 | 90.000 | -73.288 | -134.973 |
| 16.000 | 1.421 | 90.000 | -74.560 | -135.317 |
| 17.000 | -0.250 | 90.000 | -76.018 | -135.664 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 18.000 | -2.118 | 90.000 | -77.688 | -136.013 |
| 19.000 | -4.227 | 90.000 | -79.613 | -136.363 |
| 20.000 | -6.644 | 90.000 | -81.857 | -136.716 |
| 21.000 | -9.481 | 90.000 | -84.533 | -137.072 |
| 22.000 | -12.947 | 90.000 | -87.849 | -137.430 |
| 23.000 | -17.517 | 90.000 | -92.275 | -137.790 |
| 24.000 | -24.723 | 90.000 | -99.348 | -138.154 |
| 25.000 | -57.356 | -90.001 | -131.856 | 41.483 |
| 26.000 | -26.592 | -90.000 | -100.973 | 41.111 |
| 27.000 | -21.925 | -90.000 | -96.196 | 40.739 |
| 28.000 | -19.736 | -90.000 | -93.902 | 40.363 |
| 29.000 | -18.597 | -90.000 | -92.665 | 39.984 |
| 30.000 | -18.049 | -90.000 | -92.024 | 39.602 |
| 31.000 | -17.878 | -90.000 | -91.767 | 39.218 |
| 32.000 | -17.962 | -90.000 | -91.767 | 38.830 |
| 33.000 | -18.216 | -90.000 | -91.944 | 38.438 |
| 34.000 | -18.574 | -90.000 | -92.229 | 38.042 |
| 35.000 | -18.979 | -90.000 | -92.566 | 37.644 |
| 36.000 | -19.381 | -90.000 | -92.904 | 37.242 |
| 37.000 | -19.733 | -90.000 | -93.196 | 36.838 |
| 38.000 | -19.998 | -90.000 | -93.404 | 36.429 |
| 39.000 | -20.150 | -90.000 | -93.502 | 36.019 |
| 40.000 | -20.174 | -90.000 | -93.478 | 35.604 |
| 41.000 | -20.076 | -90.000 | -93.332 | 35.186 |
| 42.000 | -19.867 | -90.000 | -93.080 | 34.766 |
| 43.000 | -19.572 | -90.000 | -92.745 | 34.342 |
| 44.000 | -19.218 | -90.000 | -92.353 | 33.917 |
| 45.000 | -18.830 | -90.000 | -91.931 | 33.488 |
| 46.000 | -18.433 | -90.000 | -91.501 | 33.057 |
| 47.000 | -18.047 | -90.000 | -91.085 | 32.625 |
| 48.000 | -17.687 | -90.000 | -90.696 | 32.190 |
| 49.000 | -17.363 | -90.000 | -90.347 | 31.754 |
| 50.000 | -17.084 | -90.000 | -90.044 | 31.317 |
| 51.000 | -16.855 | -90.000 | -89.793 | 30.879 |
| 52.000 | -16.680 | -90.000 | -89.597 | 30.441 |
| 53.000 | -16.558 | -90.000 | -89.458 | 30.002 |
| 54.000 | -16.492 | -90.000 | -89.375 | 29.564 |
| 55.000 | -16.481 | -90.000 | -89.349 | 29.127 |
| 56.000 | -16.523 | -90.000 | -89.377 | 28.692 |
| 57.000 | -16.618 | -90.000 | -89.458 | 28.257 |
| 58.000 | -16.763 | -90.000 | -89.592 | 27.826 |
| 59.000 | -16.956 | -90.000 | -89.775 | 27.397 |
| 60.000 | -17.197 | -90.000 | -90.007 | 26.972 |
| 61.000 | -17.483 | -90.000 | -90.284 | 26.550 |
| 62.000 | -17.812 | -90.000 | -90.605 | 26.134 |
| 63.000 | -18.182 | -90.000 | -90.968 | 25.723 |
| 64.000 | -18.590 | -90.000 | -91.371 | 25.319 |
| 65.000 | -19.036 | -90.000 | -91.810 | 24.922 |
| 66.000 | -19.516 | -90.000 | -92.286 | 24.532 |
| 67.000 | -20.029 | -90.000 | -92.794 | 24.151 |
| 68.000 | -20.572 | -90.000 | -93.334 | 23.778 |
| 69.000 | -21.144 | -90.000 | -93.902 | 23.416 |
| 70.000 | -21.742 | -90.000 | -94.498 | 23.063 |
| 71.000 | -22.364 | -90.000 | -95.116 | 22.722 |
| 72.000 | -23.007 | -90.000 | -95.757 | 22.394 |
| 73.000 | -23.668 | -90.000 | -96.416 | 22.079 |
| 74.000 | -24.344 | -90.000 | -97.091 | 21.777 |
| 75.000 | -25.033 | -90.000 | -97.777 | 21.489 |
| 76.000 | -25.729 | -90.000 | -98.473 | 21.217 |
| 77.000 | -26.431 | -90.000 | -99.173 | 20.959 |
| 78.000 | -27.131 | -90.000 | -99.872 | 20.719 |
| 79.000 | -27.825 | -90.000 | -100.565 | 20.495 |
| 80.000 | -28.509 | -90.000 | -101.248 | 20.288 |
| 81.000 | -29.175 | -90.000 | -101.914 | 20.100 |
| 82.000 | -29.816 | -90.000 | -102.554 | 19.929 |
| 83.000 | -30.425 | -90.000 | -103.162 | 19.779 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 84.000 | -30.995 | -90.000 | -103.731 | 19.646 |
| 85.000 | -31.517 | -90.000 | -104.253 | 19.534 |
| 86.000 | -31.984 | -90.000 | -104.720 | 19.443 |
| 87.000 | -32.390 | -90.000 | -105.125 | 19.370 |
| 88.000 | -32.728 | -90.000 | -105.463 | 19.319 |
| 89.000 | -32.993 | -90.000 | -105.728 | 19.288 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |
| 0.000 | 12.511 | 90.000 | -300.000 | 0.000 |
| 1.000 | 12.473 | 90.000 | -300.000 | 0.000 |
| 2.000 | 12.356 | 90.000 | -300.000 | 0.000 |
| 3.000 | 12.160 | 90.000 | -300.000 | 0.000 |
| 4.000 | 11.885 | 90.000 | -300.000 | 0.000 |
| 5.000 | 11.530 | 90.000 | -300.000 | 0.000 |
| 6.000 | 11.092 | 90.000 | -300.000 | 0.000 |
| 7.000 | 10.571 | 90.000 | -300.000 | 0.000 |
| 8.000 | 9.962 | 90.000 | -300.000 | 0.000 |
| 9.000 | 9.265 | 90.000 | -300.000 | 0.000 |
| 10.000 | 8.473 | 90.000 | -300.000 | 0.000 |
| 11.000 | 7.584 | 90.000 | -300.000 | 0.000 |
| 12.000 | 6.590 | 90.000 | -300.000 | 0.000 |
| 13.000 | 5.482 | 90.000 | -300.000 | 0.000 |
| 14.000 | 4.252 | 90.000 | -300.000 | 0.000 |
| 15.000 | 2.887 | 90.000 | -300.000 | 0.000 |
| 16.000 | 1.367 | 90.000 | -300.000 | 0.000 |
| 17.000 | -0.333 | 90.000 | -300.000 | 0.000 |
| 18.000 | -2.247 | 90.000 | -300.000 | 0.000 |
| 19.000 | -4.433 | 90.000 | -300.000 | 0.000 |
| 20.000 | -6.982 | 90.000 | -300.000 | 0.000 |
| 21.000 | -10.056 | 90.000 | -300.000 | 0.000 |
| 22.000 | -14.000 | 90.000 | -300.000 | 0.000 |
| 23.000 | -19.763 | 90.000 | -300.000 | 0.000 |
| 24.000 | -32.870 | 90.000 | -300.000 | 0.000 |
| 25.000 | -27.536 | -90.000 | -300.000 | 0.000 |
| 26.000 | -20.633 | -90.000 | -300.000 | 0.000 |
| 27.000 | -17.594 | -90.000 | -300.000 | 0.000 |
| 28.000 | -15.891 | -90.000 | -300.000 | 0.000 |
| 29.000 | -14.873 | -90.000 | -300.000 | 0.000 |
| 30.000 | -14.274 | -90.000 | -300.000 | 0.000 |
| 31.000 | -13.950 | -90.000 | -300.000 | 0.000 |
| 32.000 | -13.815 | -90.000 | -300.000 | 0.000 |
| 33.000 | -13.810 | -90.000 | -300.000 | 0.000 |
| 34.000 | -13.893 | -90.000 | -300.000 | 0.000 |
| 35.000 | -14.029 | -90.000 | -300.000 | 0.000 |
| 36.000 | -14.193 | -90.000 | -300.000 | 0.000 |
| 37.000 | -14.367 | -90.000 | -300.000 | 0.000 |
| 38.000 | -14.532 | -90.000 | -300.000 | 0.000 |
| 39.000 | -14.682 | -90.000 | -300.000 | 0.000 |
| 40.000 | -14.810 | -90.000 | -300.000 | 0.000 |
| 41.000 | -14.917 | -90.000 | -300.000 | 0.000 |
| 42.000 | -15.003 | -90.000 | -300.000 | 0.000 |
| 43.000 | -15.076 | -90.000 | -300.000 | 0.000 |
| 44.000 | -15.143 | -90.000 | -300.000 | 0.000 |
| 45.000 | -15.210 | -90.000 | -300.000 | 0.000 |
| 46.000 | -15.286 | -90.000 | -300.000 | 0.000 |
| 47.000 | -15.380 | -90.000 | -300.000 | 0.000 |
| 48.000 | -15.498 | -90.000 | -300.000 | 0.000 |
| 49.000 | -15.647 | -90.000 | -300.000 | 0.000 |
| 50.000 | -15.835 | -90.000 | -300.000 | 0.000 |
| 51.000 | -16.064 | -90.000 | -300.000 | 0.000 |
| 52.000 | -16.342 | -90.000 | -300.000 | 0.000 |
| 53.000 | -16.670 | -90.000 | -300.000 | 0.000 |
| 54.000 | -17.055 | -90.000 | -300.000 | 0.000 |
| 55.000 | -17.499 | -90.000 | -300.000 | 0.000 |
| 56.000 | -18.007 | -90.000 | -300.000 | 0.000 |
| 57.000 | -18.582 | -90.000 | -300.000 | 0.000 |
| 58.000 | -19.230 | -90.000 | -300.000 | 0.000 |

TABLE 2 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 59.000 | -19.958 | -90.000 | -300.000 | 0.000 |
| 60.000 | -20.772 | -90.000 | -300.000 | ·0.000 |
| 61.000 | -21.682 | -90.000 | -300.000 | 0.000 |
| 62.000 | -22.702 | -90.000 | -300.000 | 0.000 |
| 63.000 | -23.848 | -90.000 | -300.000 | 0.000 |
| 64.000 | -25.146 | -90.000 | -300.000 | 0.000 |
| 65.000 | -26.631 | -90.000 | -300.000 | 0.000 |
| 66.000 | -28.357 | -90.000 | -300.000 | 0.000 |
| 67.000 | -30.416 | -90.000 | -300.000 | 0.000 |
| 68.000 | -32.969 | -90.000 | -300.000 | 0.000 |
| 69.000 | -36.357 | -90.000 | -300.000 | 0.000 |
| 70.000 | -41.515 | -90.000 | -300.000 | 0.000 |
| 71.000 | -53.753 | -90.000 | -300.000 | 0.000 |
| 72.000 | -48.364 | 90.000 | -300.000 | 0.000 |
| 73.000 | -40.717 | 90.000 | -300.000 | 0.000 |
| 74.000 | -37.003 | 90.000 | -300.000 | 0.000 |
| 75.000 | -34.629 | 90.000 | -300.000 | 0.000 |
| 76.000 | -32.941 | 90.000 | -300.000 | 0.000 |
| 77.000 | -31.677 | 90.000 | -300.000 | 0.000 |
| 78.000 | -30.701 | 90.000 | -300.000 | 0.000 |
| 79.000 | -29.934 | 90.000 | -300.000 | 0.000 |
| 80.000 | -29.330 | 90.000 | -300.000 | 0.000 |
| 81.000 | -28.854 | 90.000 | -300.000 | 0.000 |
| 82.000 | -28.484 | 90.000 | -300.000 | 0.000 |
| 83.000 | -28.202 | 90.000 | -300.000 | 0.000 |
| 84.000 | -27.998 | 90.000 | -300.000 | 0.000 |
| 85.000 | -27.861 | 90.000 | -300.000 | 0.000 |
| 86.000 | -27.784 | 90.000 | -300.000 | 0.000 |
| 87.000 | -27.763 | 90.000 | -300.000 | 0.000 |
| 88.000 | -27.794 | 90.000 | -300.000 | 0.000 |
| 89.000 | -27.874 | 90.000 | -300.000 | 0.000 |
| 90.000 | -300.000 | 0.000 | -300.000 | 0.000 |

```
FQ:
1     4.3
TL:
20.6 0.
ST:  FEED/STRUT SCATTERING
2
24   3.0    110. F    T    T    T
280.960       42.001       47.020
277.467       57.853       47.020
4
2.50
277.467       57.853       47.020
275.248       73.880       47.020
275.248       73.880       47.020
274.487       90.000       47.020
274.487       90.000       47.020
275.248      106.120       47.020
275.248      106.120       47.020
277.467      122.147       47.020
277.467      122.147       47.020
280.960      137.999       47.020
280.960      137.999       47.020
285.449      153.616       47.020
285.449      153.616       47.020
290.589      168.962       47.020
290.589      168.962       47.020
296.006      184.027       47.020
296.006      184.027       47.020
301.325      198.821       47.020
301.325      198.821       47.020
306.199      213.371       47.020
306.199      213.371       47.020
310.323      227.715       47.020
310.323      227.715       47.020
```

TABLE 2 - CONTINUED

```
313.450      241.900      47.020
313.450      241.900      47.020
315.401      255.977      47.020
315.401      255.977      47.020
316.063      270.000      47.020
316.063      270.000      47.020
315.401      284.023      47.020
315.401      284.023      47.020
313.450      298.100      47.020
313.450      298.100      47.020
310.323      312.285      47.020
310.323      312.285      47.020
306.199      326.629      47.020
306.199      326.629      47.020
301.325      341.179      47.020
301.325      341.179      47.020
296.006      355.973      47.020
296.006      355.973      47.020
290.589       11.038      47.020
290.589       11.038      47.020
285.449       26.384      47.020
285.449       26.384      47.020
280.960       42.001      47.020
PZ:
1
0.
-30.  30.  0.1
F
PP:
3
1    1
1    2
2    1
1    2
5    1
1    2
XQ:
```
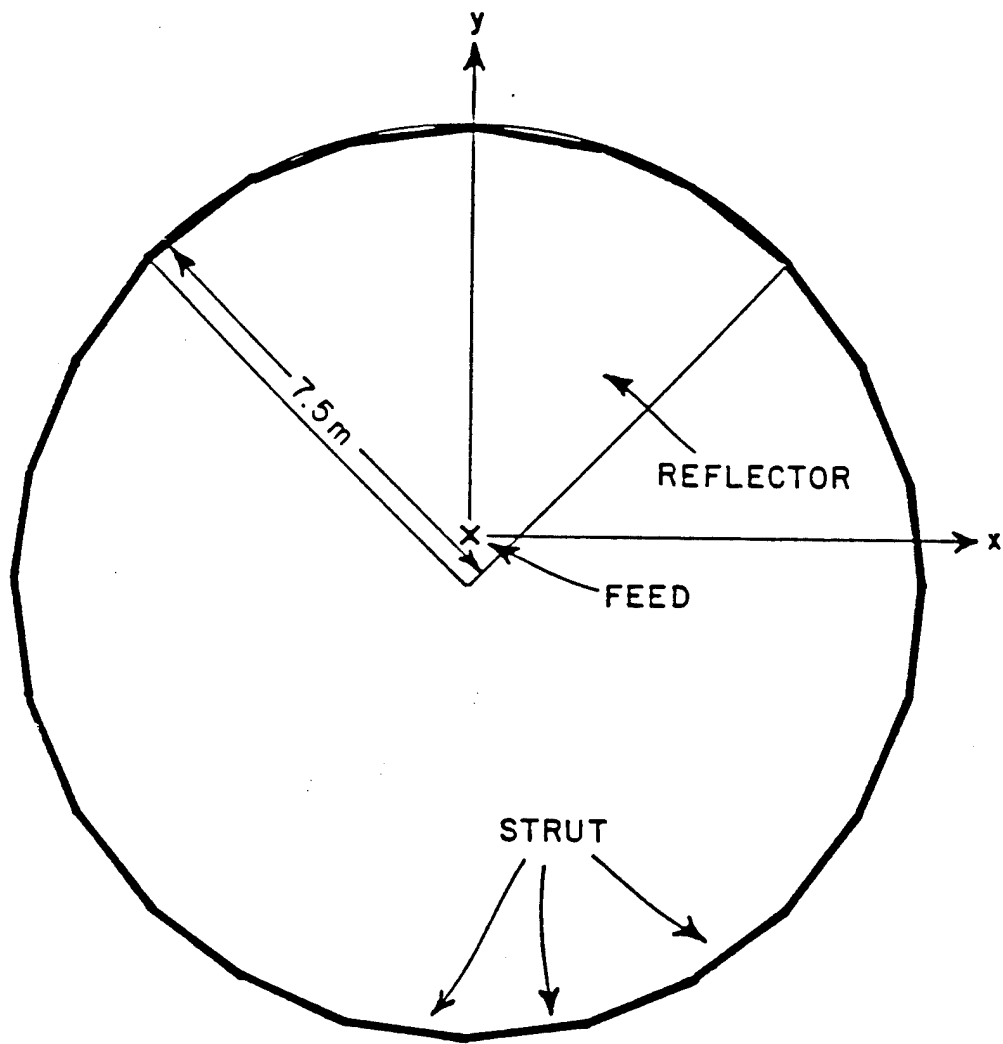
Figure 9. Geometry of a quadrant reflector with 24 struts.
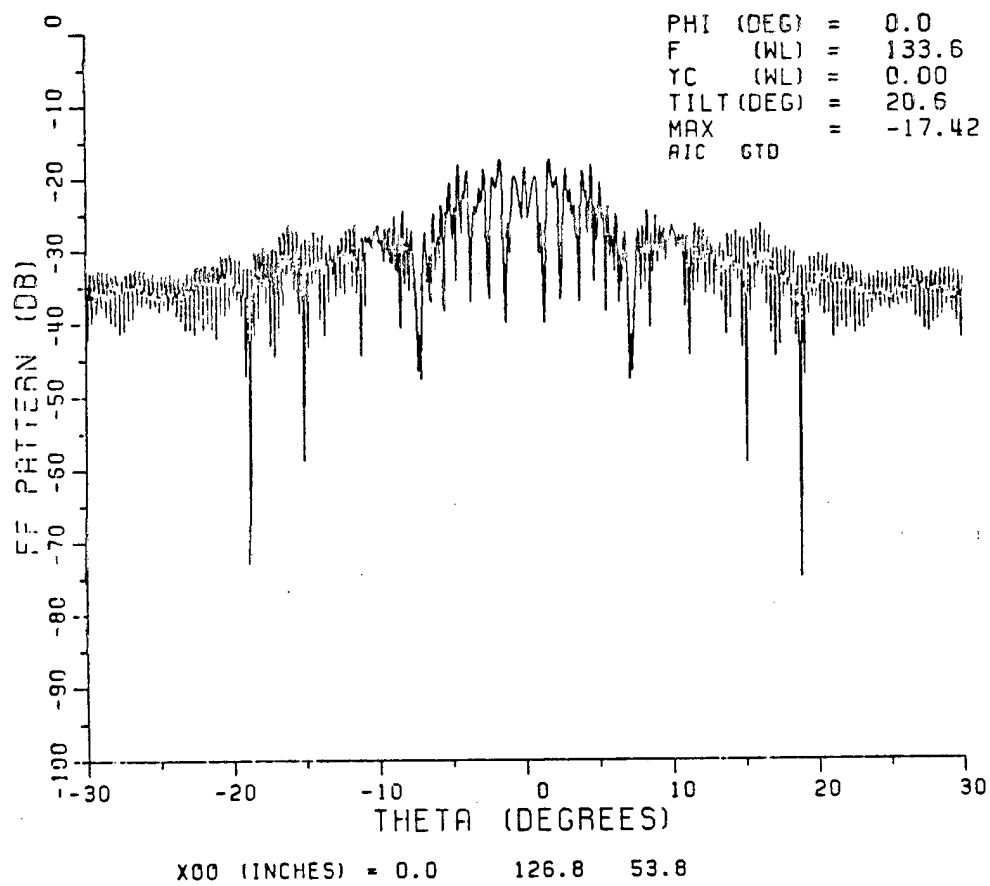
(a) feed/strut scattering fields

Figure 10. H-plane patterns of example 3.
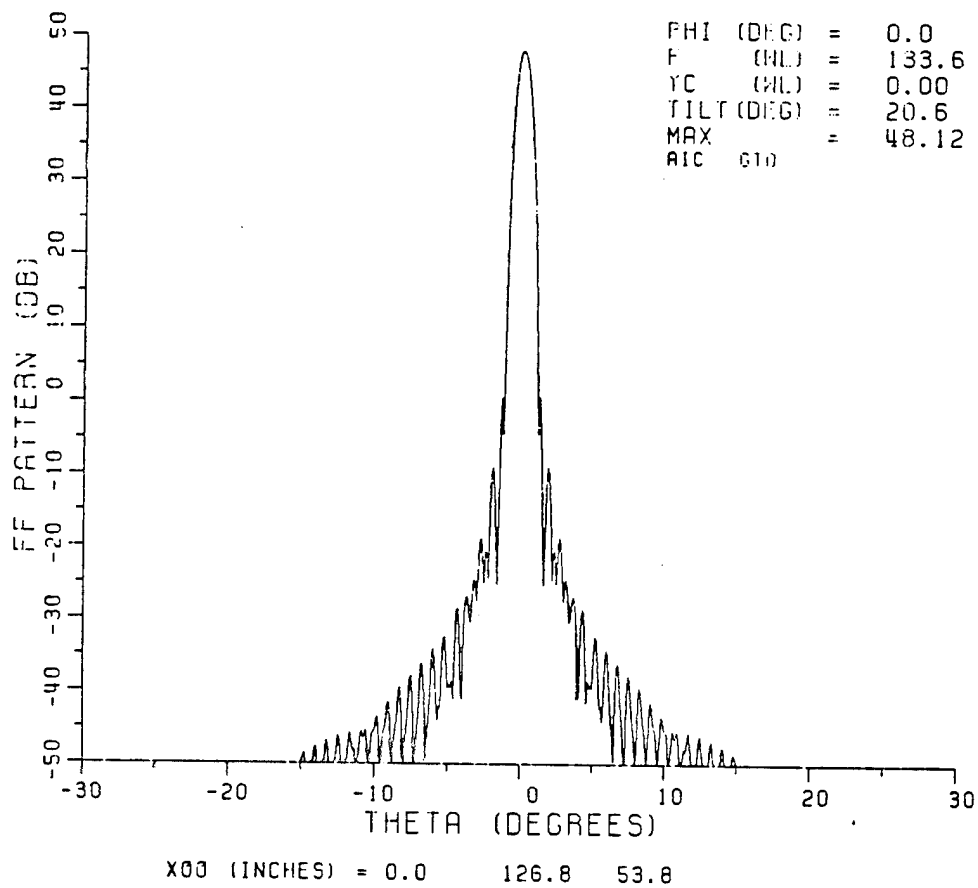
(b) reflector fields only

Figure 10. Continued.
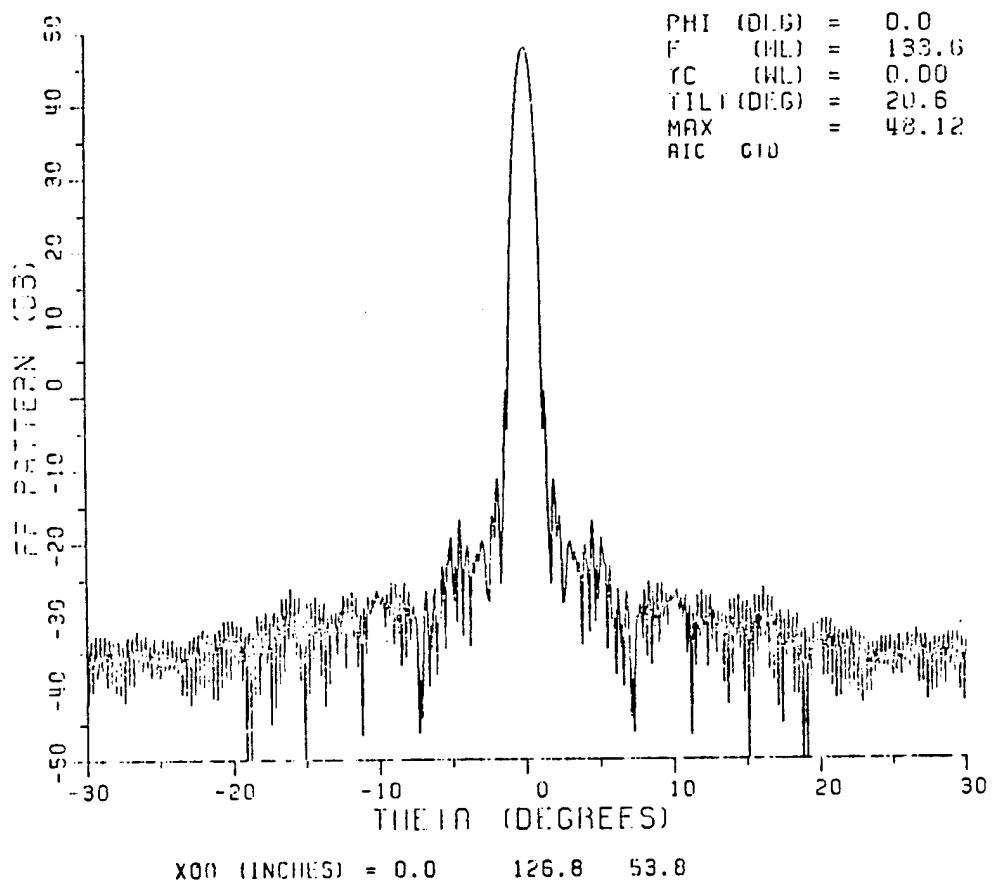
(c) total fields

Figure 10. Continued.

```
----------------------------------------------------
Command SP:   SURFACE PERTURBATION
----------------------------------------------------
```

This command enables the user to specify a periodic surface perturbation of the reflector surface. The perturbation is specified as the displacement from the ideal surface along the surface normal. Two basic ways are used to specify the surface geometry. The first way, specified by NSURF=1, consists of a series summation; the other way, specified by NSURF=2, subdivides the surface into sub sections. This command also provides for a repetitive surface in which the reflector surface perturbation repeats over NPHS periods.

1. Read:  NPHS, NSURF

   a) NPHS:  This is an integer variable which specifies the number of repetitive surface sections.

   b) NSURF:  This is an integer variable which specifies whether the series form or the subsectional form is used.

      NSURF=1:  series form
      NSURF=2:  subsectional form

An example of NPHS=6 and NSURF=1 is shown in Figure 1.


THE FOLLOWING 3 READ STATEMENTS ARE USED FOR NSURF=1 (SERIES FORM) ONLY.

2. Read:  NSFP, NSFR

   a) NSFP:  This is an integer variable which specifies the model to be used in the <u>circumferential $\phi$ direction</u> as follows:

      NSFP=0:  DPHSF=1 for no perturbations in the <u>PHI-direction.</u>

      NSFP=1:  DPHSF=$\cos(\pi \cdot DPQ)$ for corrugations.

      NSFP=2:  DPHSF=$\cos(\frac{\pi}{2} \cdot DPQ)$ for pillowing with cusps.

      NSFP=3:  DPHSF=$1-(DPQ)^2$ for pillowing with cusps.

   b) NSFR:  This is an integer variable which specifies the model to be used in the <u>RHO-direction</u> as follows:

      NSFR=1:  DRS=DRS1 + DRS2.

      NSFR=2:  DRS=|DRS1| + |DRS2|.

      NSFR=3:  DRS=RR*(DRS1 + DRS2).


<div align="center">29-1</div>

where

$$DRS1 = \sum_{1}^{NTERM} ASF(N) \cdot \sin(NSF(N) \cdot \pi \cdot RR)$$

$$DRS2 = \sum_{1}^{NTERM} BSF(N) \cdot \cos(NSF(N) \cdot \pi \cdot RR)$$

and RR=RHO/RHOMAX is the normalized radius.

3. Read:  NTERM

    a)  NTERM:  This is an integer variable which specifies the number of terms for surface perturbation used for the <u>RHO-direction.</u>

4. Read:  ASF(N), BSF(N), NSF(N)

This read statement is executed NTERM times.

    a)  ASF(N):  This is a dimensional real variable which specifies the magnitude of the $N^{th}$ sine term.

    b)  BSF(N):  This is a dimensional real variable which specifies the magnitude of the $N^{th}$ cosine term.

    c)  NSF(N):  This is an integer variable which specifies the period of the $N^{th}$ term for surface perturbation.

THE FOLLOWING READ STATEMENTS ARE USED FOR NSURF=2 (SUBSECTIONAL FORM) ONLY.

5. Read:  NMRAD,LSAME

    a)  NMRAD:  This is an integer variable which specifies the number of subsections along the <u>RHO-direction.</u>

    b)  LSAME:  This logical variable specifies whether the types of the surface perturbation in every subsection are the same or not.

6. Read: RAD(I), I=1, NMRAD + 1

   a) RAD(I): This dimensioned variable specifies the end points for the RHO-direction of each sub section.

   Note that the variable is normalized such that RAD(1)=0, and 0 < RAD(I) < 1.

THE FOLLOWING 2 READ STATEMENTS (7 AND 8) ARE USED FOR LSAME = TRUE ONLY.

7. Read: NSFP3, NSFR3

   a) NSFP3: This is an integer variable which sepcifies the type of surface perturbation along the PHI-direction.

   NSFP3 = 0:  smooth surface in the PHI-direction

   NSFP3 = 1:  $\sin(\pi \cdot PHIB)$, pillowed type

   NSFP3 = 2:  $\sin^2(\pi \cdot PHIB)$, corrugated type

   In Which PHIB is the normalized PHI-direction angle in each cell. (see Figure 2)

   b) NSFR3: This is an integer variable which specifies the type of surface perturbation along the RHO-direction.

   NSFR3=0:  constant, step type

   NSFR3=1:  $\sin(\pi \cdot RRB)$, pillowed type

   NSFR3=2:  $\sin^2(\pi \cdot RRB)$, corrugated type

   In which RRB is the normalized RHO-direction distance in each cell.

8. Read: NMRHO(I), NMPHI(I), (ASF2(I,NJ), NJ=1,NMPHI(I))

   This statement is executed NMRAD times.

   a) NMRHO(I): This dimensioned variable specifies the number of divisions along the RHO-direction in each subsection.

   b) NMPHI(I): This dimensioned variable specifies the number of divisions along the PHI-direction in each subsection. In Figure 2, an example of the subsectional form of a surface with NPHS=12 is given.

c) ASF2(I,NJ): This dimensioned variable specifies the magnitude (peak) of the surface perturbation in the cell (see Figure 2) specified by the NJ$^{th}$ PHI-division in subsection I. It is assumed that the amplitude of surface perturbation is identical for each cell along RHO-direction but may be different for each PHI-division in the same subsection.


THE FOLLOWING READ STATEMENTS ARE USED FOR LSAME = FALSE ONLY AND ARE EXECUTED NMRAD TIMES. (I=1,NMRAD)


9. Read: NMRHO((I),NMPHI(I)

The definition of these two dimensioned variables are the same as in read statement 8.


10. Read: NSFP2(I,NJ),NSFR2(I,NJ),ASF2(I,NJ)

This statement is executed NMPHI(I) times (NJ=1,NMPHI(I))

a) NSFP2(I,NJ): This dimensioned integer variable specifies the type of surface perturbation along PHI-direction in the cell specified by NJ$^{th}$ PHI-division in the I$^{th}$ subsection.

NSFP2(I,NJ)=0: smooth surface in the <u>PHI-direction</u>

NSFP2(I,NJ)=1: $\sin(\pi \cdot PHIB)$, pillowed type

NSFP2(I,NJ)=2: $\sin^2(\pi \cdot PHIB)$, corrugated type

b) NSFR2(I,NJ): This dimensioned integer variable specifies the type of surface perturbation along RHO direction in the cell specified by the NJ$^{th}$ PHI-division in I$^{th}$ subsection.

NSFR2(I,NJ)=0: constant, step type

NSFR2(I,NJ)=1: $\sin(\pi \cdot RRB)$, pillowed type

NSFR2(I,NJ)=2: $\sin^2(\pi \cdot RRB)$, corrugated type

c) ASF2(I,NJ): Same as in read statement 8. In the LSAME=FALSE case, the type and amplitude of surface perturbation of the cells along RHO-direction in each subsection are assumed to be identical.

Note: In this command, the magnitude of the surface perturbation is input in the unit which is specified in the DG: Command.
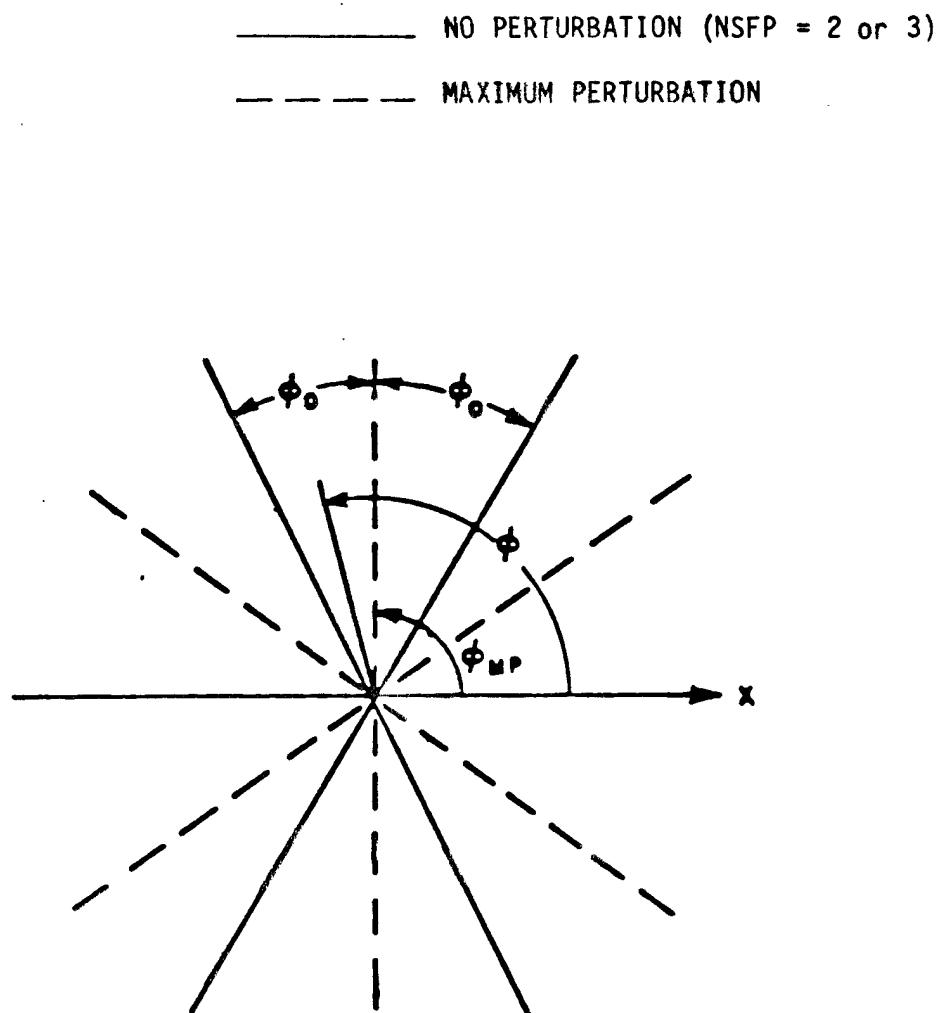
$$-1 < DPQ = \frac{\phi - \phi_{MP}}{\phi_0} < 1$$

Figure 1. Repetitive surface sections (NPHS=6 shown).

NPHS = 12,
(12 REPETITIVE SECTIONS)
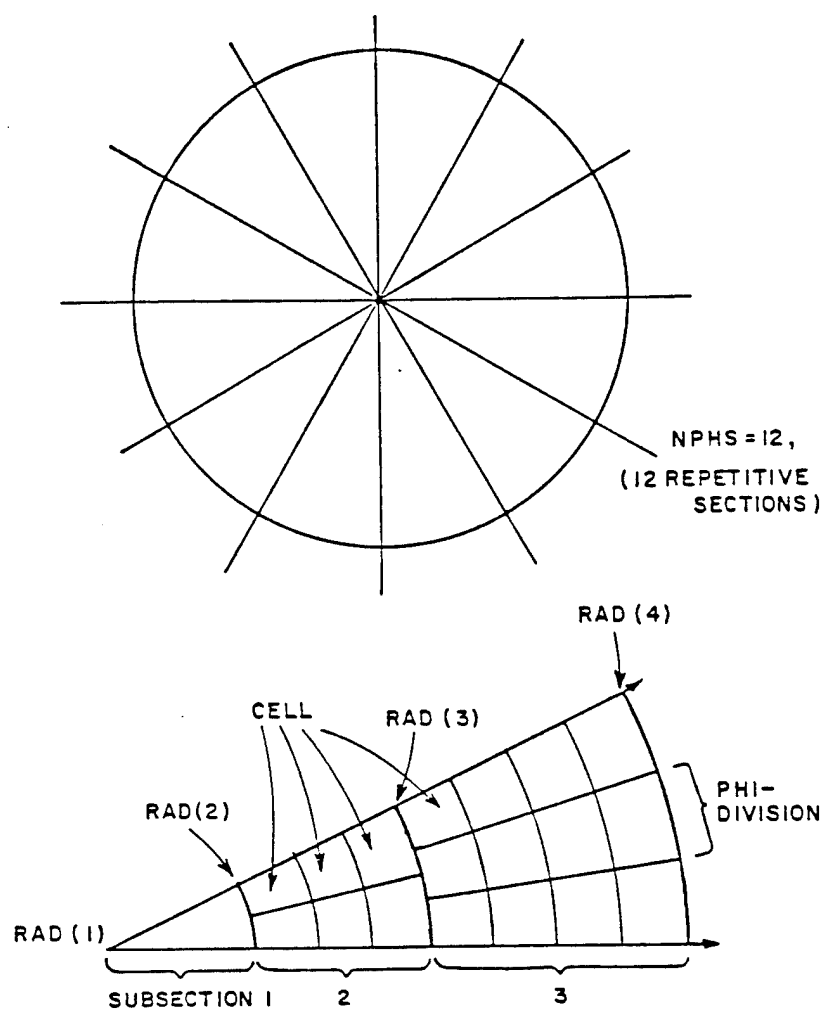
NMRAD = 3

RAD(1) = 0                NMRHO(1) = 1           NMPHI(1) = 1

RAD(2) = 0.025            NMRHO(2) = 3           NMPHI(2) = 2

RAD(3) = 0.55            NMRHO(3) = 4            NMPHI(3) = 3

RAD(4) = 1.0

Figure 2.   Example of the subsectional form of a surface.

29-6

FLOW DIAGRAM

```
                    ┌─────────────────────┐
                    │ Command:  SP:       │
                    └─────────────────────┘
                               │
                               ▼
                    ┌─────────────────────┐
                    │ Read:  NPHS,NSURF   │
                    └─────────────────────┘
                               │
                               ▼
                          ◇─────────◇
                         ╱           ╲        YES   TO SUBSECTIONAL
                        ◇ NSURF.GE.2 ? ◇  ------→ FORM FOR SURFACE
                         ╲           ╱              PERTURBATION
                          ◇─────────◇
                               │
                         NO  │ SERIES FORM FOR
                             │ SURFACE PERTURBATION
                             ▼
                    ┌─────────────────────┐
                    │ Read:  NSFP,NSFR    │
                    └─────────────────────┘
                               │
                               ▼
                    ┌─────────────────────┐
                    │ Read:  NTERM        │
                    └─────────────────────┘
                               │
                               ▼
                ┌────────────────────────────┐
                │ Loop through each term     │◀────┐
                │        N=1,NTERM           │     │
                └────────────────────────────┘     │
                               │                    │
                               ▼                    │
                ┌────────────────────────────┐     │
                │ Read:  ASF(N),BSF(N),NSF(N)│     │
                └────────────────────────────┘     │
                               │                    │
                               └────────────────────┘
```

29-7
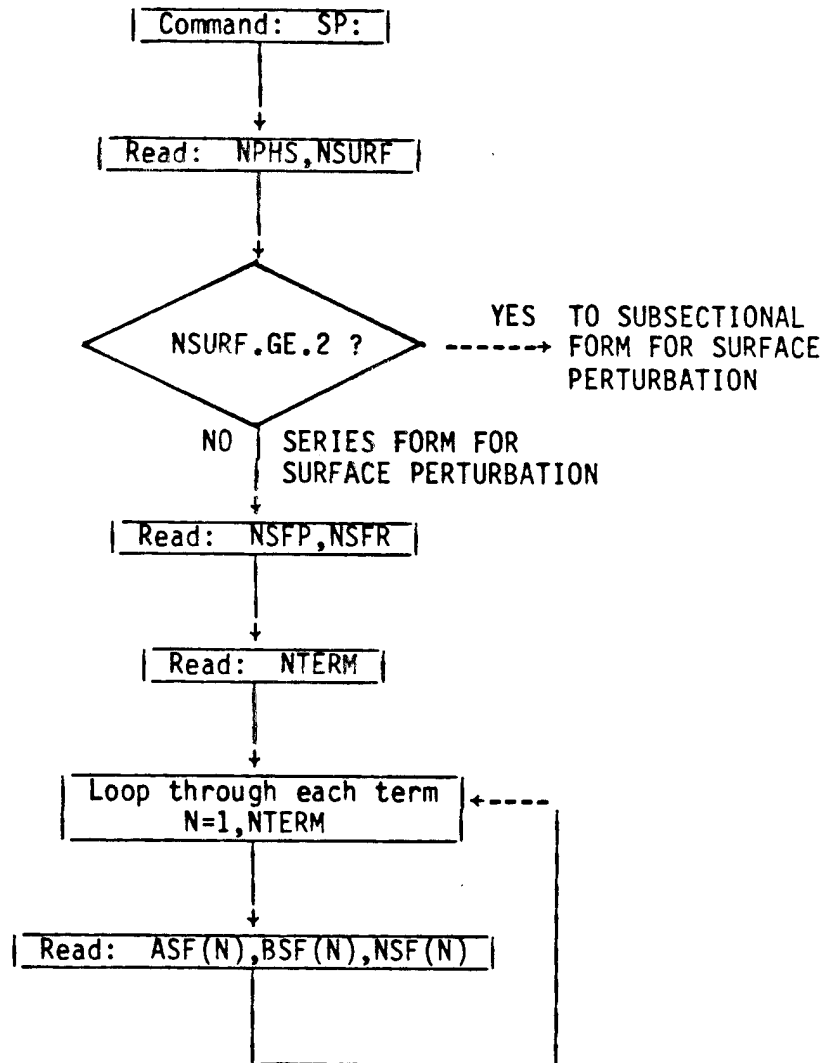
FLOW DIAGRAM (CONTINUED)

SUBSECTIONAL FORM FOR SURFACE PERTURBATION

```
                    ┌─────────────────────────┐
                    │ Read:  NMRAD,LSAME       │
                    └─────────────────────────┘
                                 │
                    ┌─────────────────────────┐
                    │ Read:  RAD(I),I=1,NMRAD+1│
                    └─────────────────────────┘
                                 │
                                 ▼
             TRUE           ╱─────────╲          FALSE
       ┌─────────────────  │  LSAME   │  ─────────────────┐
       │                    ╲─────────╱                   │
       ▼                                                  ▼
┌────────────────────┐                      ┌────────────────────────┐
│ Read:  NSFP3,NSFR3 │                      │ Loop through each       │ ◄────────┐
└────────────────────┘                      │ radial section          │          │
       │                                     │ I=1,NMRAD              │          │
       ▼                                     └────────────────────────┘          │
┌────────────────────┐                                  │                        │
│ Loop through each   │                                  ▼                        │
│ radial section I=1, │                      ┌────────────────────────┐          │
│ NMRAD              │ ◄───                  │ Read:                   │          │
└────────────────────┘                      │ NMRHO(I),NMPHI(I)       │          │
       │                                     └────────────────────────┘          │
       ▼                                                  │                        │
┌──────────────────────────────┐                          ▼                        │
│ Read:  NMRHO(I),NMPHI(I),     │            ┌────────────────────────┐          │
│ (ASF2(I,NJ),NJ=1,NMPHI(I)     │            │ Loop through each       │          │
└──────────────────────────────┘            │ phi-direction division   │ ◄───    │
                                             │ in Ith radial section   │          │
                                             │ NJ=1,NMPHI(I)          │          │
                                             └────────────────────────┘          │
                                                         │                        │
                                                         ▼                        │
                                             ┌────────────────────────┐          │
                                             │ Read:  NSFP2(I,NJ),     │          │
                                             │ NSFR2(I,NJ),            │          │
                                             │ ASF2(I,NJ)              │          │
                                             └────────────────────────┘          │
                                                         │                        │
                                                         └────────────────────────┘
```
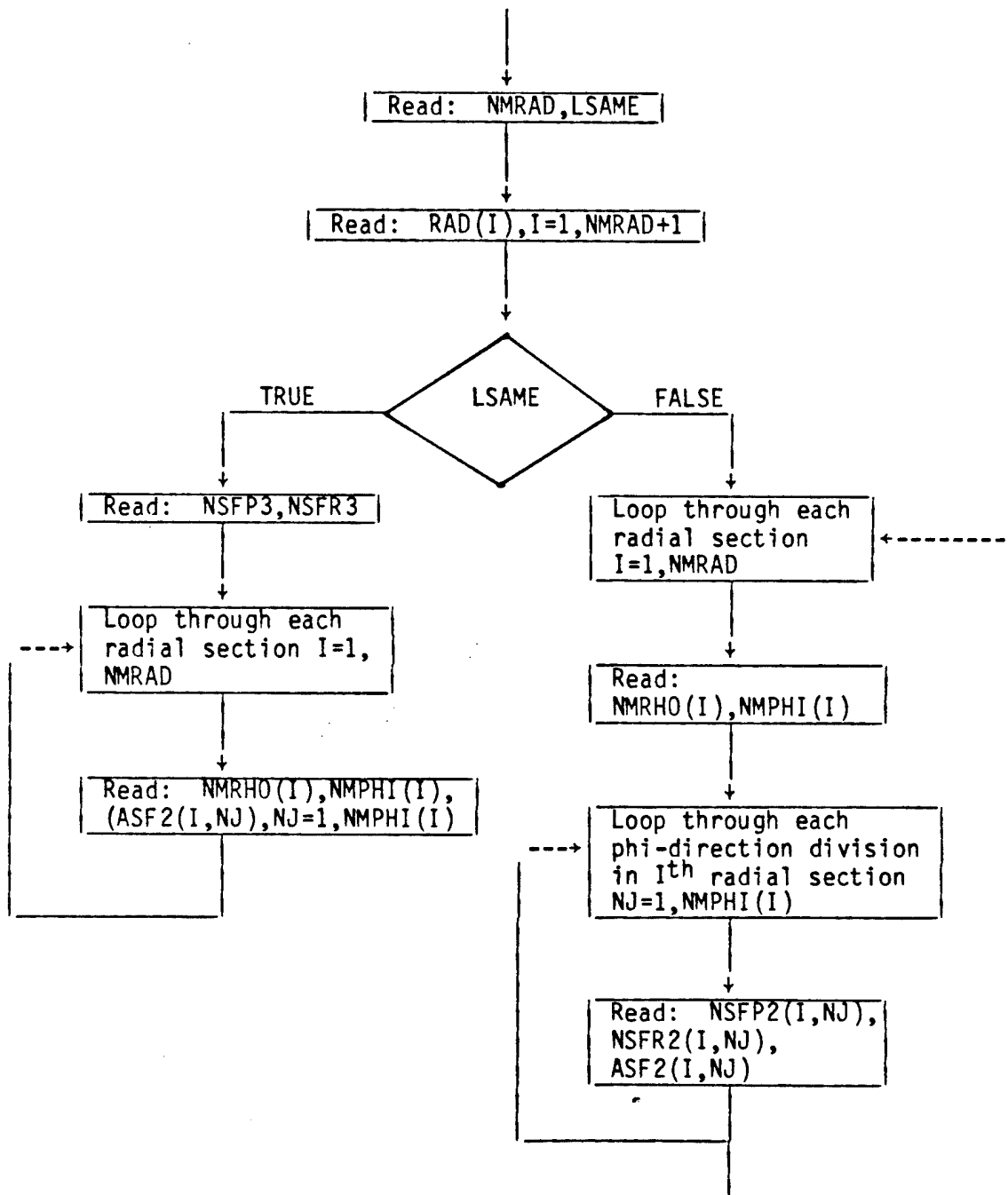
EXAMPLE 1:

This example demonstrates how the SP: Command is used to calculate the radiation pattern of a PI-shaped reflector. The geometry of the reflector is given in Figure 3a. In this example, the surface perturbation is the case of the subsectional form (NSURF=2) and the pillowed type (NSFR3=NSFP3=1). The input data are given in Table 1. The calculated principal plane patterns are given in Figure 4.

# TABLE 1

## INPUT DATA FOR CALCULATING PATTERNS OF THE PI-SHAPED REFLECTOR

```
CM:        ***** SP35.DAT *****
CM: EXAMPLE OF SURFACE PERTURBATION
CM: SUBSECTIONAL FORM PERTURBATION
CM:       NSFP3=1, NSFR3=1 FOR
CE:         SIN(PI*X) MODEL
DG:
1
3    20.3    0.11    0.11    0.    20
  10.66     9.51
   9.19    10.95
   8.20    11.71
   7.15    12.38
   6.04    12.95
   4.89    13.43
   3.70    13.80
   2.48    14.07
   1.25    14.24
   0.00    14.29
  -1.25    14.24
  -2.48    14.07
  -3.70    13.80
  -4.89    13.43
  -6.04    12.95
  -7.15    12.38
  -8.20    11.71
  -9.19    10.95
 -10.66     9.51
   0.00    -1.15
TO: AIC ONLY
F    90.    1.
F    32    32    1    48
F    F     F     F    0
T    T     F    0.8
T    F     F     F     F     F    0.    0.
F    F    0.    0.
FD: NHORN=2 AND LSI=TRUE
2    T
T
11.4  1.3  11.4  1.3  0.  90.  F
FQ:
1       35.
AF:
1
0.   0.   20.3
0.   0.    0.
1.   0.
90.  20.  90.
SP:
48   2
4    T
0. 0.173 0.383 0.602 1.
  1   1
  1   1   0.
  7   1   0.067
  7   2   0.034    0.034
 13   3   0.0417  0.0417 0.0417
```

```
PZ:
2
0. -90.
-90. 90.  0.5
F
PP:
1
1   1
1   2
XQ:
```
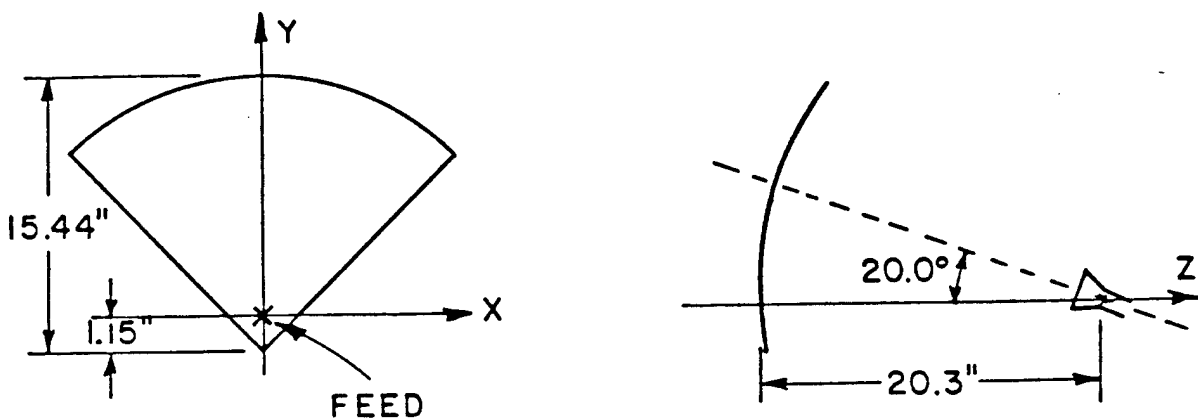
FREQUENCY = 35.0 GHz
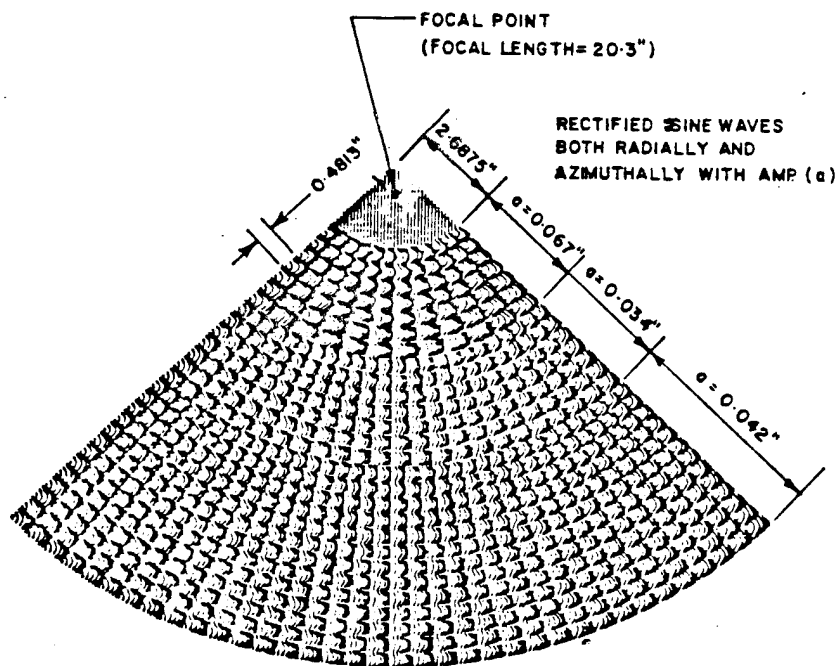


Figure 3a. Geometry of a PI-shaped reflector with pillowed surface.
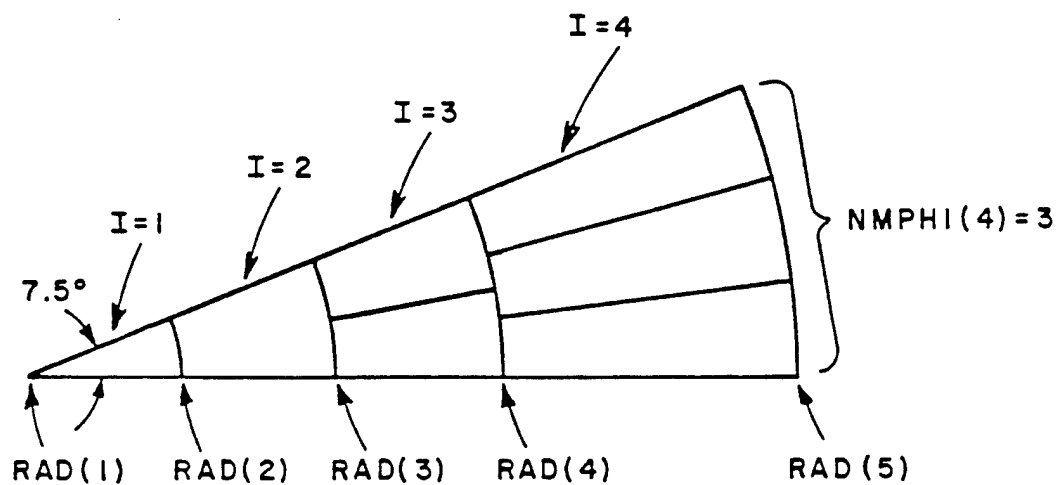
NPHS   =48:

NMRAD = 4

RAD(1)=0., RAD(2)=0.173, RAD(3)=0.383, RAD(4)=0.602, RAD(5)=1.0

| I | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| NMPHI(I) | 1 | 1 | 2 | 3 |
| NMRHO(I) | 1 | 7 | 7 | 13 |

Figure 3b.  One of the repetitive sections for the reflector of
            Figure 3a.
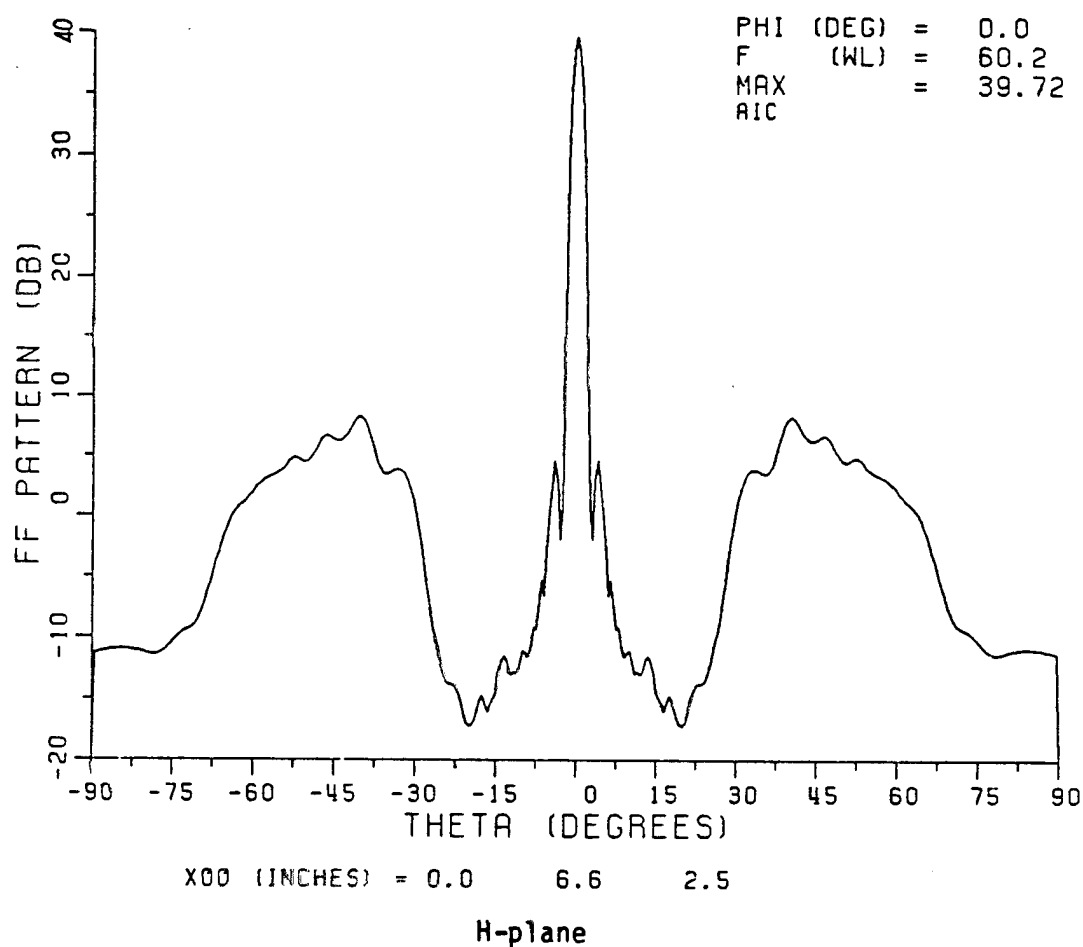
Figure 4. Radiation patterns of a pi-shaped reflector with the subsectional form (NSURF=2) and the pillowed type (NSFR3=NSFP3=1) surface perturbation.
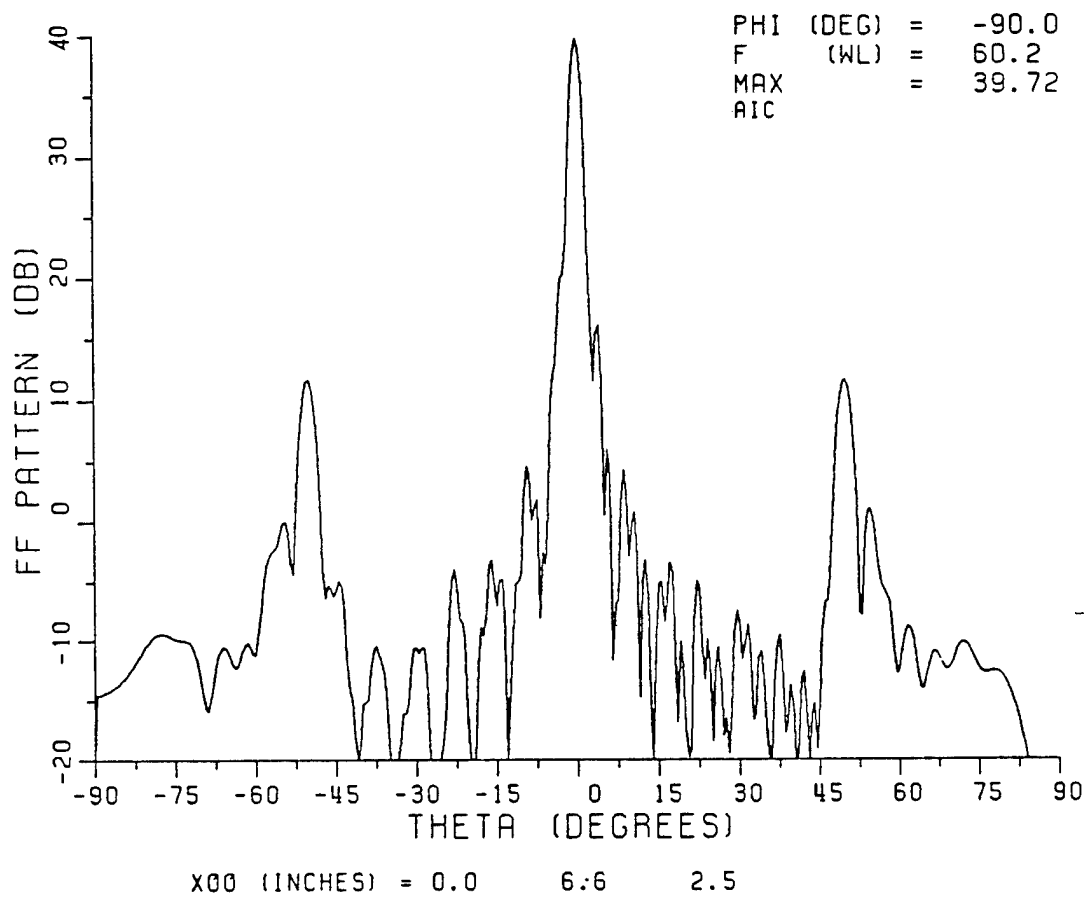
E-plane

Figure 4. Continued.

```
----------------------------------------------------------
```
## Command SD: NONPERIODIC SURFACE DISTORTION
```
----------------------------------------------------------
```

This command enables the user to specify the nonperiodic surface distortion of the reflector surface. The distortion is specified as the displacement along the normal from the ideal surface. It provides a capability to input the amplitude and the location of distortion. More than one location of distortion can be input. The type of distortion can be chosen from four functions for each input distortion point. The effective range of this distortion can also be specified. A system of equations is then solved to obtain the required amplitude corresponding to each distortion function.

1. Read: NPSD

    a) NPSD: This integer variable specifies the number of distortion regions on the reflector surface.

2. Read: RHOSD(NSD),PHISD(NSD),PV(NSD),RRAN(NSD),NTYPSD(NSD).

    This statement is executed NPSD times. All inputs are in units.

    a) RHOSD(NSD): This variable specifies the RHO-coordinate of the center of the NSD$^{th}$ distortion region.

    b) PHISD(NSD): This variable specifies the PHI-coordinate of the center of the NSD$^{th}$ distortion region.

    c) PV(NSD): This variable specifies the total amplitude of distortion at the center of the NSD$^{th}$ distortion region.

    d) RRAN(NSD): This variable specifies the effective range (radius) of the NSD$^{th}$ distortion region with respect to the center of this region.

    e) NTYPSD(NSD): This integer variable specifies the type of distortion for the NSD$^{th}$ distortion region.

        NTYPSD(NSD) = 0: constant, step type surface distortion

        NTYPSD(NSD) = 1: Circular cone type surface distortion

        NTYPSD(NSD) = 2: $\cos(\pi*X/2)$ type surface distortion

        NTYPSD(NSD) = 3: $\cos^2(\pi*X/2)$ type surface distortion

    in which X is the distance between a given point and the center of the NSD$^{th}$ distortion region.

# BLOCK DIAGRAM FOR NONPERIODIC SURFACE DISTORTION

```
                        | Command SD: |
                              |
                              |
                        | LSFD=.TRUE. |
                              |
                              |
                        | Read:  NPSD |
                              |
                              |
------------------------| DO  NSD=1,NPSD |
|                             |
|                             |
|   | Read:  RHOSD(NSD),PHISD(NSD),PV(NSD), |
|   |        RRAN(NSD),NTYPSD(NSD)           |
|                             |
|                             |
--------------------| CONTINUE |
```
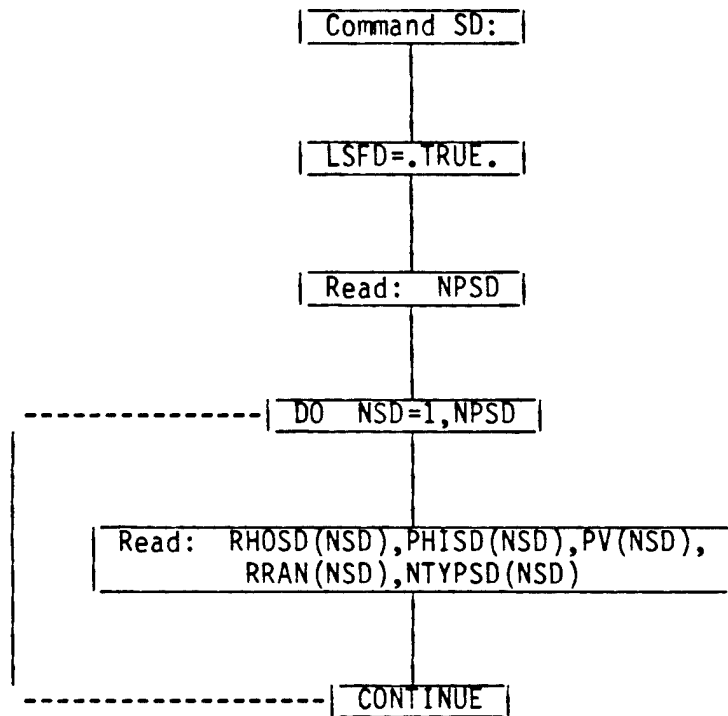
---------------------------------------------------------------
COMMAND PS: PLOT SURFACE PERTURBATION
---------------------------------------------------------------

This command enables the user to plot the surface perturbation or surface distortion (displacement along the normal to the ideal surface) along radial lines at any phi angles.

1. Read: NPHISF

   a) NPHISF: This integer variable is used to specify the desired number of the plots for the surface perturbation.

2. Read: (ANPS(L), L=1, NPHISF)

   a) ANPS(L): This dimensional real variable is input in degrees and defines the phi angle of the radial line for the $L^{th}$ plot of surface perturbation.

3. Read: RRI, RRF, DRR

   These three variables are all input in units.

   a) RRI: This real variable defines the initial radial coordinate for the plot of the surface perturbation.

   b) RRF: This real variable defines the final radial coordinate for the plot of the surface perturbation.

   c) DRR: This real variable defines the increment in radial coordinate for the plot of the surface perturbation.

Note that the data for the plots are stored in the write unit #15 and currently the plot routine PSURF is used to plot it.

BLOCK DIAGRAM FOR PLOT SURFACE PERTURBATION

```
            | Command:  PS |
                    |
                    |
             | Read:  NPHISF |
                    |
                    |
    | Read:  (ANPS(L), L=1, NPHISF) |
                    |
                    |
       | Read:  RRI, RRF, DRR |
```
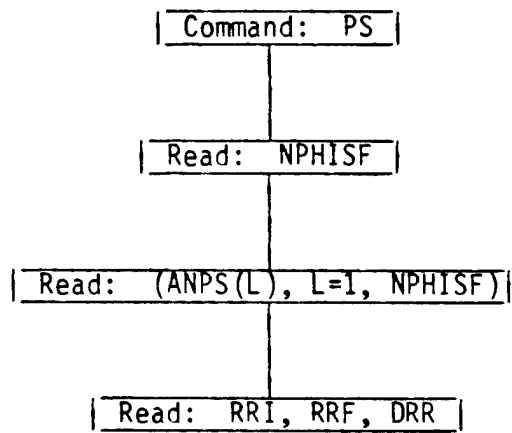
---
Command PF:   PLOT FEED PATTERN
---

     This command enables the user to plot feed patterns.  The plot data for feed patterns is output on Unit #10.


BLOCK DIAGRAM FOR PLOTS OF FEED PATTERNS

```
                    | Command PF: |
                           |
                           |
                           |
                    | Read: NPHIP |
                           |
                           |
                           |
              | Read: ANPF(L), L=1,NPHIP |
                           |
                           |
                           |
                | Read:  PFI,PFF,DPF |
                           |
                           |
                           |
                    | Read: NPFD |
                           |
                           |
                           |
                | DO NI=1,NPFD |
         _____|
        |                  |
        |                  |
        |         | Read: IFD(NI) |
        |                  |
        |                  |
        |                  |
        -------------| CONTINUE |
```

32-1
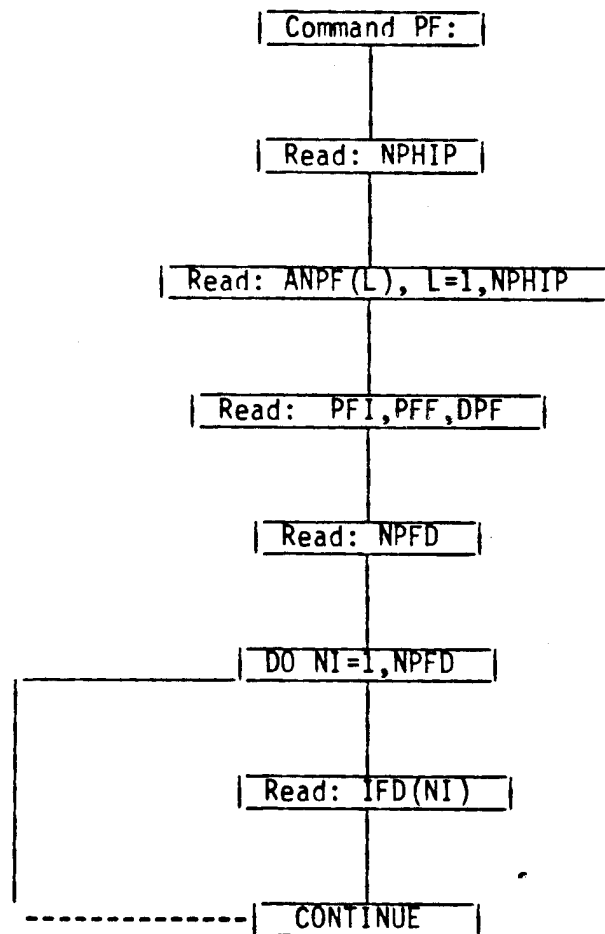
1. READ:   NPHIP

    a) NPHIP:  This integer variable specifies the number of PHI-cuts to
       be plotted for the feed pattern.  Feed plot data is output on
       Unit #10 Presently NPHIP<10.


2. READ:   (ANPF(L),L=1,NPHIP)

    a) ANPF(L):  This dimensioned real variable defines the Lth PHI-cut
       for the feed pattern plot.


3. READ:   PFI,PFF,DPF

    a) PFI,PFF,DPF:  These real variables define the initial angle,
       final angle, and increment in angle, respectively,for the feed
       pattern plots.  (in degrees)


4. READ:   NPFD

    a) NPFD:  This integer variable specifies the number of feed pattern
       plots for each PHI-cut.  1<NPFD<3.


5. READ:   IFD(NI)

    This read statement is executed NPFD times.

    a) IFD(NI):  This integer specifies the format to be plotted as
       follows:

                    1 magnitude of feed pattern.
          IFD = < 2 dB value of feed pattern.
                    3 phase of feed pattern in degrees.

```
-----------------------------------------------------------
Command PY:  PLOT YSUM
-----------------------------------------------------------
```

This command enables the user to plot YSUM data (Y-integrations used for AI).  Plot data for YSUM is output on Unit #11.

BLOCK DIAGRAM FOR PLOTS OF YSUM DATA

```
                        |  Command PY:  |
                                |
                                |
                        |  Read:  NPYM  |
                                |
                                |
                        |  DO NI=1,NPYM  |
            |-----------------
            |
            |       | Read: NYM(NI),IYM(NI) |
            |                   |
            |                   |
            |------------|  CONTINUE  |
```
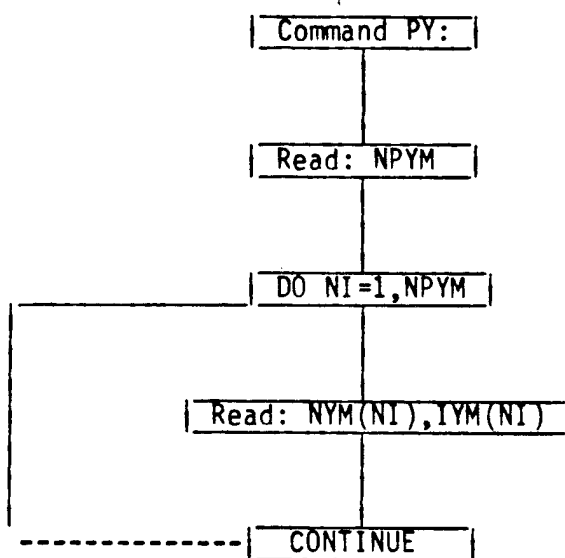
1. READ: NPYM

   a) NPYM: This integer variable specifies the number of YSUM plots.


2. READ: NYM(NI),IYM(NI)

   This read statement is executed NPYM times.

   a) NYM(NI): This integer specifies the polarization component to be plotted as follows:

$$N = \begin{cases} 1 & \text{X-component of YSUM} \\ 2 & \text{Y-component of YSUM} \end{cases}$$

   b) IYM(NI): This integer specifies the format to be plotted as follows:

$$I = \begin{cases} 1 & \text{magnitude of YSUM.} \\ 2 & \text{dB value of YSUM.} \\ 3 & \text{phase of YSUM in degrees.} \end{cases}$$

```
------------------------------------------------
Command PC:  PLOT CSUM
------------------------------------------------
```

This command enables the user to plot the CSUM, which is the integrand of the x-integration at one specified far field pattern point. From that plot, the user can see how the YSUM's contribute to the total pattern value.

1. Read:  NCSUM

   a) NCSUM:  This integer variable is used to specify the number of CSUM plots desired.  If NCSUM < 0, no plot is desired. If NCSUM=4, real and imaginary plots for both x and y components will be obtained.  If NCSUM=2, real and imaginary plots of the y component will be obtained.

Note that the pattern point desired is specified by the PZ: command and should be restricted to one point for each run.  The data for the plots are stored in the write unit #12 and currently the plot routine PCSUM is used to plot it.

BLOCK DIAGRAM FOR PLOT CSUM

```
          | Command:  PC |
                 |
                 |
                 |
          | Read:  NCSUM |
```

34-1
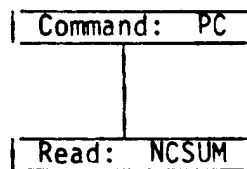
---
Command PG:   PLATE GEOMETRY
---

This command enables the user to define the geometry of the flat plate structures to be considered. One call to this command defines one plate. The number of plates in the structure are automatically counted by the number of calls to this command.

1. Read:  MEP(MP), MEPD(MP)

   a) MEP(MP):  This dimensioned integer variable is used to define the number of corners (or edges) on the $MP^{th}$ plate.

   b) MEPD(MP):  This dimensional integer variable is used to define the number of edges on the $MP^{th}$ plate from which the diffracted fields are calculated.

2. Read:  IXPD(MP,MED), MED=1, MEPD(MP)
   This statement is skipped if MEP(MP) = MEPD(MP)

   a) IXPD(MP,MED):  This dimensional integer variable specifies the indicies of the edges on the $MP^{th}$ plate from which the diffracted fields are calculated.

3. Read:  (XQ(MP,ME,N), N=1,3)

   a) XQ(MP,ME,N):  This triply dimensioned real variable is used to specify the location of the $ME^{th}$ corner of the $MP^{th}$ plate. It is input on a single line with the real numbers being the x,y,z coordinates of the corner, in the specified coordinate system, which corresponds to N=1,2,3, respectively, in the array. For example, the array will contain the following for plate 1 and corner 2 located at x=2., y=4., z=6.,:

   XQ(1,2,1)=2.
   XQ(1,2,2)=4.
   XQ(1,2,3)=6.

   This data is input as:  2.,4.,6.

   Presently:  1<MP<5
               3<ME<7

# BLOCK DIAGRAM FOR PLATE GEOMETRY

```
                    | Command PG: |
                          |
              | Set plate logic true |
              |   LPLA=.TRUE.        |
                          |
             | Up plate counter by one |
             |     MPX=MPX+1           |
                          |
                | Write plate number |
                |     MP=MPX          |
                          |
          |Read:  MEP(MP),MEPD(MP)|
          |       MEX=MEP(MP)     |
          |       MEXD=MEPD(MP)   |
                          |
                        /    \
              F        /      \
      - - - - - - - - < IF(MEXD.LT.MEX)? >
      |                 \      /
      |                  \    /       |
      |                   \  /   -->| DO MED=1,MEX |
      |                    \/       |
      |                  T |        |
      |                    |     --| IXPD(ME,MED)=MED |
      | | READ:  IXPD(MP,MED),MED=1,MEXD |
      |                    |
      |- - - - - - -| Do ME=1,MEX |
      |                    |
      |  | Read:   (XQ(MP,ME,N),N=1,3) |
      |                    |
      |- - - - - - -| Continue |
                          |
      | Translate and rotate the coordinates |
                          |
             | Change units to meters |
                          |
           | Write new and old coordinates |
           |          and units           |
```

35-2
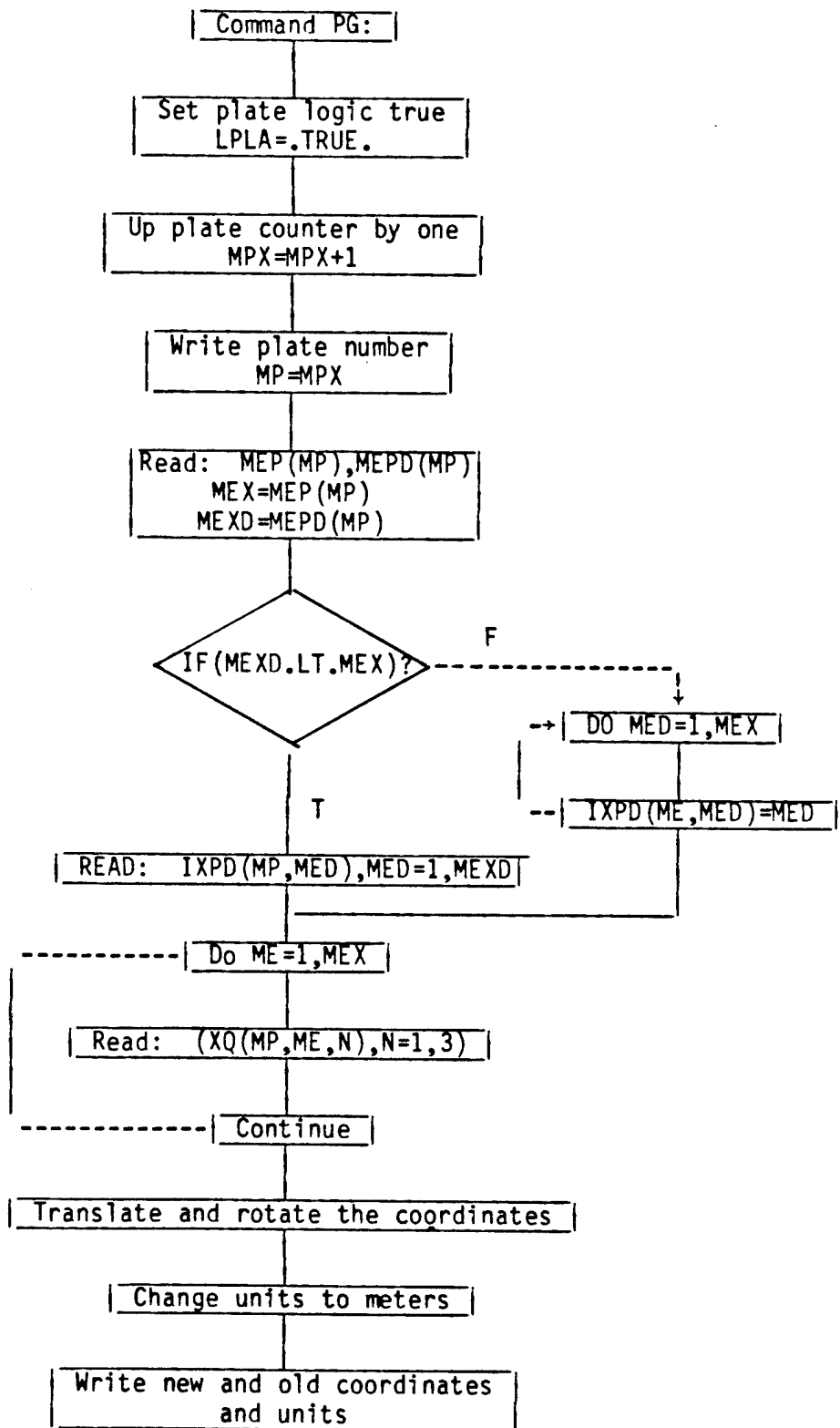
---------------------------------------------------------
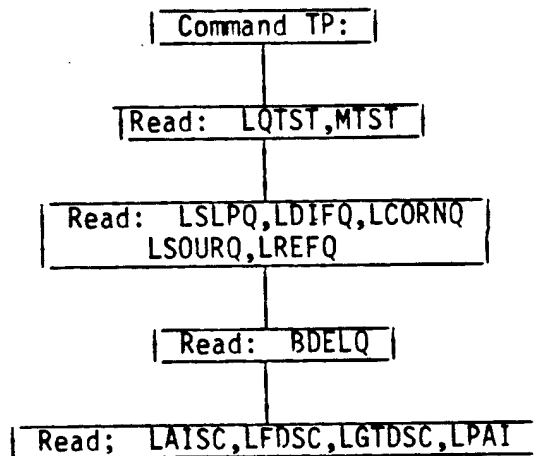Command TP:  PLATE SCATTERING
---------------------------------------------------------


     This command provides the user a test version for the plate
scattering, and also enables the user to specify whether a GTD source,
a primary feed source or a YSUM source model will be used for the plate
scattering.  The GTD and YSUM source models are illustrated in Figures
1, 2, and 3.

     NOTE:  When thge YSUM source model is used (LAISC=true), the upper
and lower edges of the scatterer should lie outside the projected
aperture as shown in Figure 1.

     NOTE:  The user should be aware of the relative position of the
plate to the reflector antenna, then choose the input variables LAISC,
LGTDSC, and THETAX properly.  See the notes associated with the read
statements for these variables in this manual.




BLOCK DIAGRAM FOR TEST PLATE


                        | Command TP: |
                               |
                               |
                 |Read:  LQTST,MTST |
                               |
                               |
             | Read:  LSLPQ,LDIFQ,LCORNQ |
             |        LSOURQ,LREFQ        |
                               |
                               |
                   | Read:   BDELQ |
                               |
                               |
        | Read;  LAISC,LFDSC,LGTDSC,LPAI |


                              36-1

1.  Read:  LQTST,MTST

    a)  LQTST:  This logical variable is used to determine if a
        test run is desired.  If LQTST is set true, only a single
        test source which is specified by MTST is considered.  This
        option is very helpful for the debugging.  If LQTST is set
        false, the code will run through all of the sources as
        usual.  (normally set false)

    b)  MTST:  If LQTST is true, this integer variable is used to
        specify which single source is under consideration.  When
        the GTD source model is employed, MTST is the index of
        a single reflector rim segment.  When the YSUM model is
        employed, MTST is the index of a single YSUM value.  If
        LOTST is false, this input will be neglected.

2.  Read:  LSLPQ,LDIFQ,LCORNQ,LSOURQ,LREFQ:

    These logical variables allow certain GTD diffraction terms for
    the plate scatterer to be suppressed for test purposes.

    a)  LSLPQ:  This is a proposed logical variable used to
        indicate if slope diffraction from the plate is desired.
        Since the current development does not include the slope
        diffraction from the plate yet, this input is neglected by
        the code.

    b)  LDIFQ:  This logical variable is used to tell the code
        whether or not edge diffraction is desired during the
        computation.  (normally set true)

    c)  LCORNQ:  This logical variable is used to tell the code
        whether or not corner diffraction is desired during the
        computation.  (normally set true)

    d)  LSOURQ:  This logical variable is used to tell the code
        whether or not the direct source field from the reflector
        is desired during the computation.  (normally set true)

    e)  LREFQ:  This logical variable is used to tell the code
        whether or not the reflected field from the plate is
        desired during the computation.  (normally set true)

3. Read: BDELQ

   a) BDELQ: This real variable is used in the PLATE subroutine
      to adjust the bounds for corner diffraction from plates.
      Normally, 0.5<BDELQ<0.8; smaller values in this range may
      improve run times, at some loss of some accuracy. Presently
      BDELQ=0.8, unless this command is used.


4. Read: LAISC,LFDSC,LGTDSC,LPAI

   a) LAISC: This logical variable indicates if the YSUM source
      model, shown in Figure 1, is desired for the scattered
      field. The YSUM source model should be chosen for the
      scattered field if part of a plate extends into the AI
      region of the reflector as shown in Figure 4.

   b) LFDSC: This logical variable indicates if the primary feed
      source model, shown in Figure 3, is desired for the
      scattered field.

   c) LGTDSC: This logical variable indicates if the GTD source
      model, shown in Figure 2, is desired for the scattered
      field. The GTD source model can be used for the scattered
      field if the plate does not extend into the AI region of the
      reflector as shown in Figure 4; or at pattern angles wider
      than 60 degrees.

   d) LPAI: This is a logical variable which is set true if the
      plate effect in the AI region needs to be considered. If
      the plate effect in the AI region is expected to be small
      and can be neglected. LPAI can be set false to save
      computer run time. (normally set true)

## CONTROL of AI(YSUM), PRIMARY FEED
## and GTD SOURCE MODELS for PLATE SCATTERING

The logical variables LAISC, LFDSC and LGTDSC are chosen
as shown in the following table:

| LAISC | LFDSC | LGTDSC | DECISION |
|-------|-------|--------|----------|
| F | F | F | No scattered field is desired. The program will skip the whole scattering loop. |
| F | F | T | Scattered field is calculated by using the GTD source model in all regions. |
| T | F | F | Scattered field is calculated by using the YSUM source model in all regions. |
| T | F | T | Scattered field is calculated by the GTD source model in the GTD region and by the YSUM model in the AI region. The regions are shown in Figure 4a. |
| F | T | F | Scattered field is calculated by using the primary feed source model in all regions. |
| F | T | T | Scattered field is calculated by using the primary feed and GTD source models in all regions. |
| T | T | F | Scattered field is calculated by using the primary feed and AI source models in all regions. |
| T | T | T | Scattered field is calculated by the GTD source model in the GTD region, YSUM source model in the AI region, and primary feed source model in all regions. |

NOTE: If part of a plate extends into the AI region, then an AI source model is recommended for pattern angles near the plate shadow as shown in Figure 4b, i.e., either the YSUM source model, if satisfied, or the 2-D array source model in conjunction with the Basic Scattering Code. The GTD source model is recommended for pattern angles wider than 60 degrees even if the plate shadow extends beyond 60 degrees from the refelctor axis as shown in Figure 4c. If both LAISC and LGTDSC are set true, the AI/GTD switching angle THETAX in the TO: command should be chosen carefully such that the above criterions will be met.
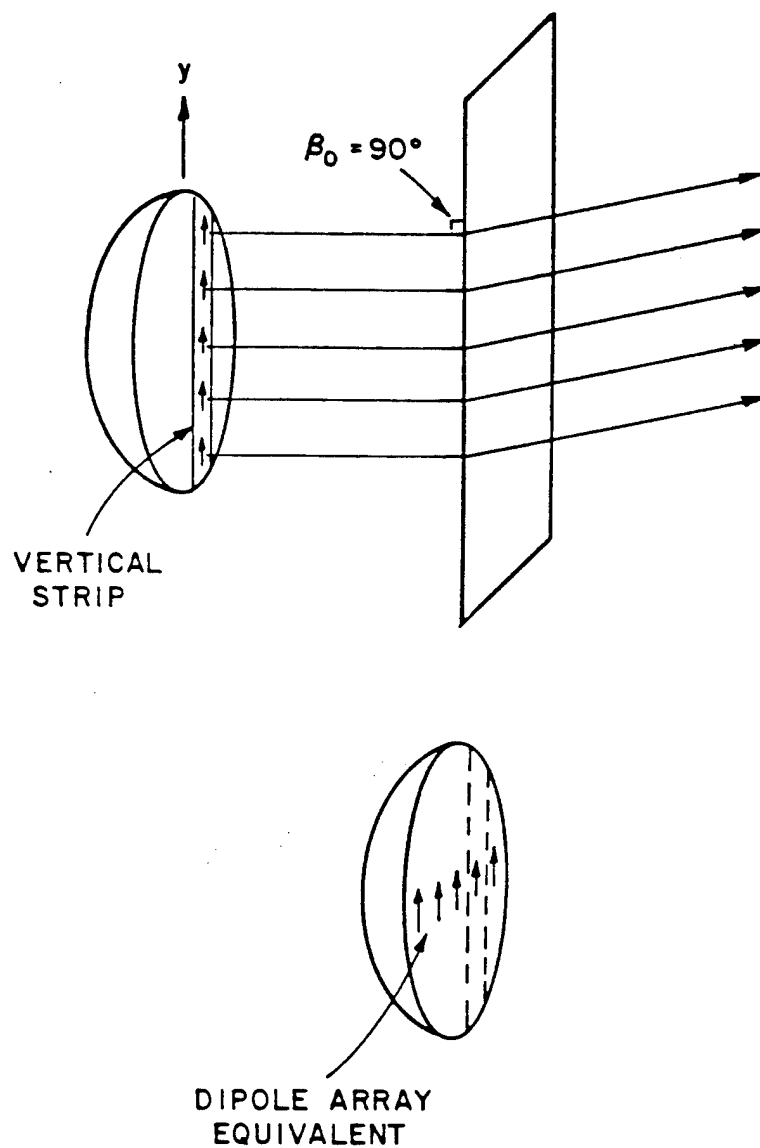
Y $-$ SUM SOURCE MODEL FOR TRANSVERSE
PLANE ( WORST CASE SCAT.TERING )

Figure 1. Y-sum source model for transverse plane (worst case scattering).

TYPICAL RIM
SEGMENT
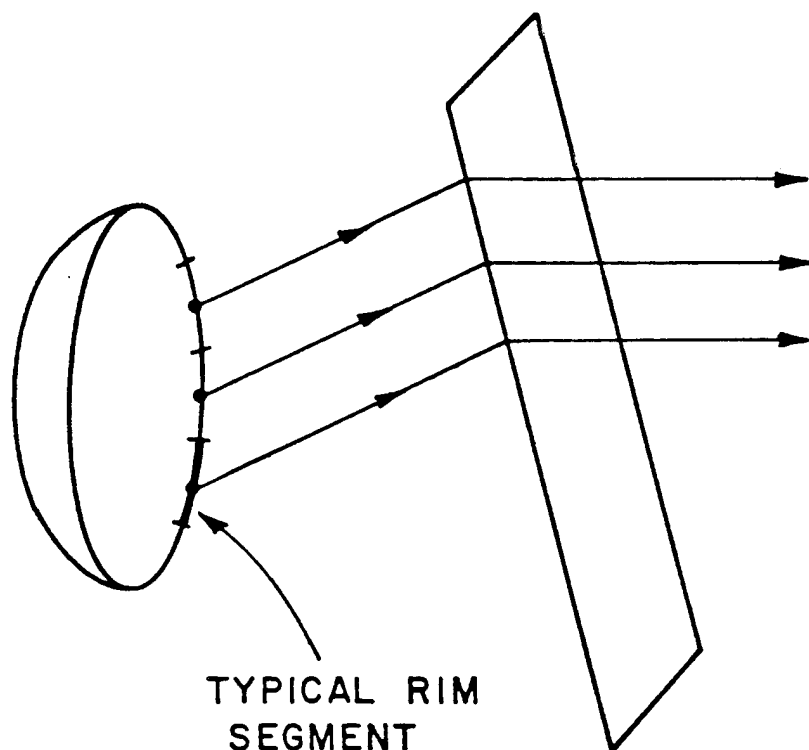
GTD SOURCE MODEL

Figure 2. GTD source model.

PLATE

PRIMARY FEED

REFLECTOR
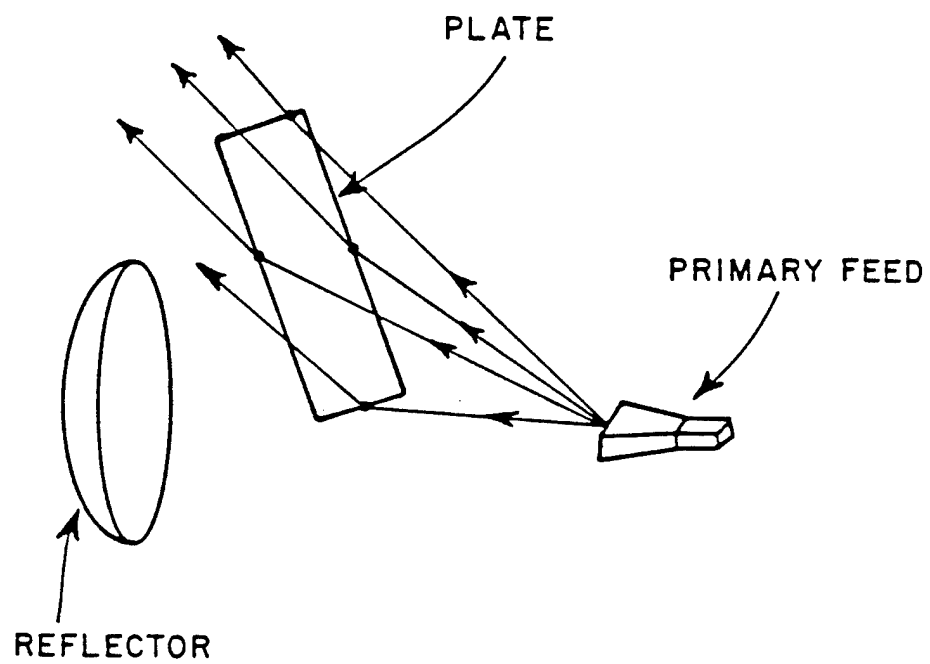
Figure 3. Primary feed source model.
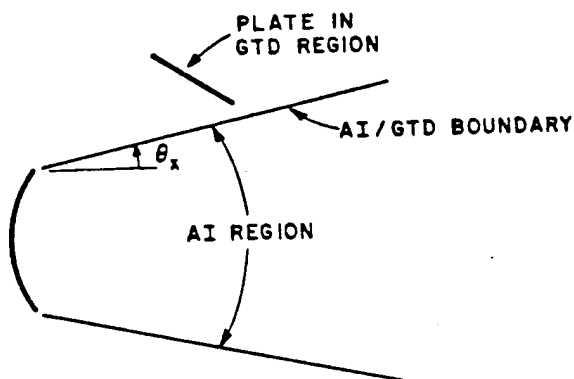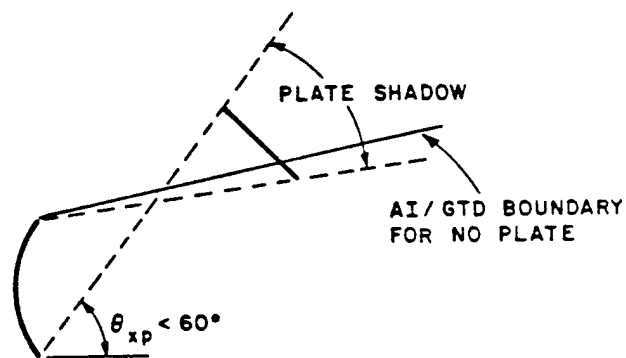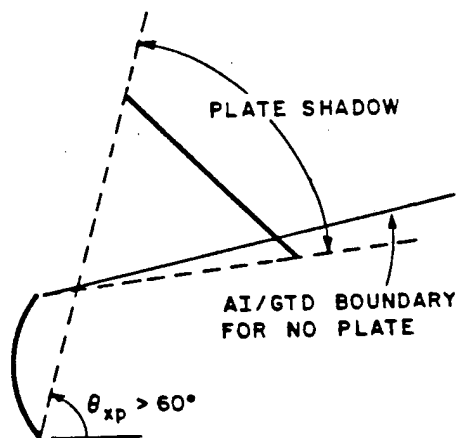
Figure 4. Regions for each source model. a) Plate in the GTD region. The code uses the YSUM model in the AI region and the GTD region, unless LAISC=F or LGTDSC=F. b) Part of plate extends into the AI region adjust AI/GTD switching angle $\theta_x > \theta_{xp}$ (if LAISC=LGTDSC=T). c) Part of plate extends into the AI region and plate shadow extends beyond 60° adjust AI/GTD switching angle $\theta_x=60$ (if LAISC=LGTDSC=T).

36-9

Example 1:

     This example illustrates how the PG: command and the TP command can be used to calculate the effects of a shroud on the spillover fields of a reflector. The reflector in this example is identical to the one in example 1 of the DG: command of this supplement, i.e., a Cassegrain reflector antenna. The calculation of the subreflector pattern will not be repeated here. Only the H-plane main reflector pattern is shown here for the purpose of comparison. The geometry of the main reflector with the shroud is shown in Figure 5. The flat plate simulation of the shroud is shown in Figure 6 for the H-plane pattern calculation. Note that only the spillover fields of the feed and the edge diffracted fields from one edge of the plate (edge 2) are included in order to model the edge of the shroud. The input data for the flat plate simulation are given in Table 1.

# TABLE 1

## INPUT DATA FOR CALCULATING SHROUD EFFECTS

```
CM:            *** SHROUD.DAT ***
CM: EXAMPLE FOR CALCULATING SHROUD EFFECTS
CM: 2 FEET DISH WITH CONICAL HORN FEED
CM:          AND SUBREFLECTOR
CM:        DIAMETER OF HORN=1.2"
CM:     FLARE ANGLE OF HORN= 14.9 DEG.
CM:       DIAMETER OF SUBREFLECTOR=2.92"
CE:
DG:
1
3.  8.0   0.5   0.5   24.   0
FQ:
1 38.0
FD:
0     T
T    0    T    1    90.  0.   F
3   0.  45.  90.
181
       0.000      0.967      8.395    -123.560    -71.142
       1.000      1.082      7.778     -90.530    -46.993
       2.000      1.435      6.196    -105.978    -64.826
       3.000      1.956      4.145    -124.967   -161.925
       4.000      2.550      2.168    -104.320   -176.473
       5.000      3.121      0.639    -127.370    -17.627
       6.000      3.593     -0.267    -125.602    -66.099
       7.000      3.874      1.039    -131.047   -101.178
       8.000      3.889      1.309    -131.781     32.642
       9.000      3.826      1.901    -123.943     47.900
      10.000      3.706      2.646    -121.349     50.985
      11.000      3.559      3.342    -121.309     51.826
      12.000      3.417      3.801    -123.247     51.218
      13.000      3.311      3.897    -128.319     46.439
      14.000      3.265      3.610    -141.442    -21.585
      15.000      3.285      3.030    -129.788   -104.446
      16.000      3.364      2.336    -125.346   -109.515
      17.000      3.486      1.728    -124.394   -109.249
      18.000      3.619      1.362    -125.649   -106.278
      19.000      3.734      1.312    -129.968    -96.757
      20.000      3.807      1.571    -138.812    -35.522
      21.000      3.825      2.058    -130.777     41.253
      22.000      3.785      2.646    -126.023     52.585
      23.000      3.692      3.184    -124.197     56.532
      24.000      3.571      3.539    -124.273     58.040
      25.000      3.450      3.611    -126.224     57.473
      26.000      3.353      3.363    -130.966     51.498
      27.000      3.303      2.844    -141.060     -6.898
      28.000      3.313      2.178    -132.596    -86.091
      29.000      3.380      1.528    -127.823    -93.677
      30.000      3.481      1.041    -126.098    -93.787
      31.000      3.602      0.827    -126.349    -90.846
      32.000      3.717      0.920    -128.378    -84.060
      33.000      3.804      1.285    -132.875    -65.842
      34.000      3.848      1.840    -136.967      0.533
      35.000      3.842      2.470    -131.361     49.657
      36.000      3.785      3.049    -127.601     63.333
      37.000      3.683      3.458    -125.944     69.386
      38.000      3.561      3.609    -125.737     72.702
      39.000      3.437      3.449    -126.908     74.274
      40.000      3.336      2.986    -129.847     72.987
      41.000      3.275      2.290    -136.095     61.282
      42.000      3.266      1.480    -141.183    -34.845
```

TABLE 1 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 43.000 | 3.303 | 0.695 | -132.360 | -69.240 |
| 44.000 | 3.384 | 0.076 | -128.360 | -72.981 |
| 45.000 | 3.494 | -0.283 | -126.469 | -72.019 |
| 46.000 | 3.614 | -0.334 | -125.858 | -69.028 |
| 47.000 | 3.725 | -0.092 | -126.295 | -64.255 |
| 48.000 | 3.812 | 0.403 | -127.713 | -56.796 |
| 49.000 | 3.859 | 1.071 | -130.245 | -43.746 |
| 50.000 | 3.858 | 1.823 | -133.655 | -16.565 |
| 51.000 | 3.815 | 2.567 | -134.714 | 31.628 |
| 52.000 | 3.733 | 3.198 | -128.632 | 34.120 |
| 53.000 | 3.619 | 3.654 | -129.169 | 49.483 |
| 54.000 | 3.483 | 3.859 | -129.912 | 65.635 |
| 55.000 | 3.333 | 3.773 | -130.870 | 83.004 |
| 56.000 | 3.183 | 3.386 | -131.924 | 102.447 |
| 57.000 | 3.054 | 2.719 | -132.968 | 124.615 |
| 58.000 | 2.960 | 1.831 | -133.746 | 149.794 |
| 59.000 | 2.910 | 0.807 | -133.946 | 176.851 |
| 60.000 | 2.909 | -0.248 | -133.464 | -156.837 |
| 61.000 | 2.944 | -1.234 | -132.502 | -133.476 |
| 62.000 | 3.016 | -2.064 | -131.367 | -113.527 |
| 63.000 | 3.117 | -2.672 | -130.256 | -96.411 |
| 64.000 | 3.237 | -3.021 | -129.255 | -81.386 |
| 65.000 | 3.364 | -3.104 | -128.414 | -67.817 |
| 66.000 | 3.480 | -2.933 | -127.677 | -55.402 |
| 67.000 | 3.576 | -2.536 | -127.063 | -43.794 |
| 68.000 | 3.652 | -1.944 | -126.558 | -32.776 |
| 69.000 | 3.704 | -1.196 | -126.146 | -22.209 |
| 70.000 | 3.729 | -0.330 | -125.814 | -11.989 |
| 71.000 | 3.722 | 0.616 | -125.551 | -2.038 |
| 72.000 | 3.674 | 1.612 | -125.368 | 7.701 |
| 73.000 | 3.597 | 2.623 | -125.205 | 17.288 |
| 74.000 | 3.491 | 3.619 | -125.076 | 26.756 |
| 75.000 | 3.357 | 4.575 | -124.969 | 36.131 |
| 76.000 | 3.189 | 5.479 | -124.873 | 45.436 |
| 77.000 | 2.986 | 6.314 | -124.777 | 54.678 |
| 78.000 | 2.759 | 7.052 | -124.671 | 63.841 |
| 79.000 | 2.509 | 7.679 | -124.541 | 72.907 |
| 80.000 | 2.238 | 8.180 | -124.384 | 81.844 |
| 81.000 | 1.927 | 8.584 | -124.189 | 90.634 |
| 82.000 | 1.596 | 8.842 | -123.951 | 99.207 |
| 83.000 | 1.247 | 8.946 | -123.668 | 107.512 |
| 84.000 | 0.881 | 8.882 | -123.342 | 115.498 |
| 85.000 | 0.478 | 8.688 | -122.995 | 123.116 |
| 86.000 | 0.050 | 8.328 | -122.580 | 130.361 |
| 87.000 | -0.396 | 7.768 | -122.130 | 137.179 |
| 88.000 | -0.860 | 6.998 | -121.649 | 143.556 |
| 89.000 | -1.372 | 6.066 | -121.139 | 149.490 |
| 90.000 | -1.917 | 4.907 | -120.607 | 154.975 |
| 91.000 | -2.553 | 6.360 | -119.561 | 149.359 |
| 92.000 | -2.723 | 3.219 | -119.100 | 154.484 |
| 93.000 | -3.628 | -1.867 | -118.570 | 159.414 |
| 94.000 | -4.930 | -2.807 | -117.988 | 163.773 |
| 95.000 | -5.342 | -2.113 | -119.006 | 14.223 |
| 96.000 | -5.136 | -7.017 | -119.575 | 11.252 |
| 97.000 | -5.721 | -14.781 | -120.251 | 7.634 |
| 98.000 | -6.789 | -17.769 | -120.990 | 3.514 |
| 99.000 | -6.998 | -17.620 | -121.738 | -1.495 |
| 100.000 | -6.706 | -23.629 | -122.498 | -7.952 |
| 101.000 | -7.180 | -33.313 | -123.265 | -15.983 |
| 102.000 | -8.206 | -38.625 | -123.962 | -25.286 |
| 103.000 | -8.414 | -40.311 | -124.421 | -35.901 |
| 104.000 | -8.031 | -47.854 | -124.568 | -47.619 |
| 105.000 | -8.375 | -59.257 | -124.432 | -59.398 |
| 106.000 | -9.279 | -65.931 | -124.062 | -69.938 |
| 107.000 | -9.243 | -68.409 | -123.514 | -78.792 |
| 108.000 | -8.531 | -77.667 | -122.914 | -86.314 |

TABLE 1 - CONTINUED

| | | | |
|---|---|---|---|
| 109.000 | -8.764 | -91.749 | -122.402 | -92.715 |
| 110.000 | -10.037 | -99.590 | -122.080 | -97.749 |
| 111.000 | -10.157 | -97.850 | -121.902 | -101.743 |
| 112.000 | -8.716 | -105.381 | -121.907 | -105.560 |
| 113.000 | -8.371 | -124.184 | -122.205 | -109.706 |
| 114.000 | -10.167 | -142.012 | -122.904 | -113.894 |
| 115.000 | -12.893 | -140.526 | -123.966 | -118.228 |
| 116.000 | -11.275 | -132.610 | -125.354 | -124.327 |
| 117.000 | -8.882 | -152.178 | -127.265 | -135.534 |
| 118.000 | -8.674 | 176.150 | -129.672 | -156.274 |
| 119.000 | -11.018 | 144.553 | -131.250 | 170.848 |
| 120.000 | -16.360 | 136.975 | -130.103 | 139.199 |
| 121.000 | -14.048 | 168.498 | -127.669 | 121.204 |
| 122.000 | -8.873 | 148.539 | -125.472 | 111.054 |
| 123.000 | -6.900 | 115.103 | -124.009 | 103.660 |
| 124.000 | -7.629 | 81.177 | -123.420 | 97.961 |
| 125.000 | -11.642 | 55.124 | -123.611 | 94.174 |
| 126.000 | -17.757 | 80.400 | -124.398 | 91.478 |
| 127.000 | -11.696 | 97.430 | -125.828 | 87.092 |
| 128.000 | -8.319 | 66.666 | -128.393 | 76.540 |
| 129.000 | -7.550 | 26.988 | -132.577 | 48.681 |
| 130.000 | -8.407 | -13.563 | -133.668 | -11.746 |
| 131.000 | -10.847 | -46.476 | -129.444 | -46.544 |
| 132.000 | -14.359 | -51.909 | -126.347 | -57.913 |
| 133.000 | -12.254 | -36.506 | -124.508 | -62.648 |
| 134.000 | -8.573 | -52.061 | -123.699 | -66.192 |
| 135.000 | -7.354 | -80.123 | -124.016 | -70.324 |
| 136.000 | -8.537 | -111.299 | -125.739 | -75.666 |
| 137.000 | -12.240 | -139.766 | -129.443 | -84.774 |
| 138.000 | -17.524 | -141.113 | -136.273 | -118.016 |
| 139.000 | -13.566 | -126.662 | -134.006 | 159.660 |
| 140.000 | -7.798 | -154.211 | -127.847 | 138.908 |
| 141.000 | -4.055 | 170.506 | -124.578 | 131.720 |
| 142.000 | -2.084 | 134.759 | -123.171 | 126.337 |
| 143.000 | -1.716 | 97.941 | -123.434 | 120.790 |
| 144.000 | -2.773 | 57.522 | -125.637 | 114.372 |
| 145.000 | -4.811 | 10.509 | -130.873 | 102.363 |
| 146.000 | -7.185 | -42.313 | -139.789 | 25.751 |
| 147.000 | -10.738 | -97.049 | -130.656 | -36.306 |
| 148.000 | -18.547 | 166.132 | -125.977 | -41.811 |
| 149.000 | -9.374 | 52.100 | -123.711 | -42.512 |
| 150.000 | -3.399 | 9.179 | -123.007 | -43.618 |
| 151.000 | -0.622 | -27.103 | -123.981 | -47.454 |
| 152.000 | 0.511 | -64.930 | -127.491 | -58.417 |
| 153.000 | 1.028 | -106.879 | -135.410 | -112.772 |
| 154.000 | 1.773 | -150.367 | -128.474 | 164.551 |
| 155.000 | 2.655 | 169.735 | -123.284 | 148.255 |
| 156.000 | 2.982 | 133.899 | -121.297 | 140.515 |
| 157.000 | 2.346 | 98.230 | -121.502 | 134.297 |
| 158.000 | 0.955 | 56.745 | -124.004 | 126.351 |
| 159.000 | 0.169 | 6.055 | -130.912 | 105.643 |
| 160.000 | 1.100 | -43.307 | -132.447 | -2.658 |
| 161.000 | 2.352 | -83.833 | -124.636 | -28.435 |
| 162.000 | 3.025 | -121.222 | -121.514 | -35.673 |
| 163.000 | 3.603 | -160.794 | -120.732 | -37.259 |
| 164.000 | 3.121 | 159.640 | -122.514 | -40.537 |
| 165.000 | 3.186 | 121.981 | -127.833 | -41.935 |
| 166.000 | 3.582 | 91.819 | -149.018 | 159.574 |
| 167.000 | 3.284 | 67.246 | -126.600 | 145.894 |
| 168.000 | 1.840 | 42.724 | -121.729 | 145.149 |
| 169.000 | -0.566 | 9.675 | -120.158 | 146.036 |
| 170.000 | -1.587 | -38.024 | -120.284 | 148.279 |
| 171.000 | 0.356 | -77.615 | -123.674 | 150.846 |
| 172.000 | 2.090 | -100.324 | -135.072 | 156.949 |
| 173.000 | 2.296 | -115.463 | -126.944 | -33.554 |
| 174.000 | 0.664 | -132.219 | -128.067 | 13.192 |

TABLE 1 - CONTINUED

| | | | |
|---|---|---|---|
| 175.000 | -2.889 | -161.415 | -121.323 | 65.996 |
| 176.000 | -3.472 | 137.785 | -101.095 | -75.063 |
| 177.000 | 1.231 | 102.703 | -111.180 | -79.016 |
| 178.000 | 4.529 | 91.185 | -100.571 | -103.429 |
| 179.000 | 6.297 | 86.110 | -91.446 | 58.021 |
| 180.000 | 6.886 | 84.371 | -113.952 | 37.563 |
| 0.000 | 0.967 | 8.395 | -123.048 | -63.194 |
| 1.000 | 1.143 | 7.866 | -38.904 | 157.565 |
| 2.000 | 1.641 | 6.510 | -27.562 | 159.050 |
| 3.000 | 2.331 | 4.825 | -21.518 | 159.313 |
| 4.000 | 3.054 | 3.313 | -17.959 | 159.588 |
| 5.000 | 3.679 | 2.256 | -16.041 | 159.893 |
| 6.000 | 4.121 | 1.739 | -15.457 | 160.273 |
| 7.000 | 4.453 | 0.130 | -17.175 | 158.870 |
| 8.000 | 4.440 | 0.398 | -17.402 | 159.474 |
| 9.000 | 4.220 | 1.176 | -20.014 | 160.257 |
| 10.000 | 3.857 | 2.275 | -26.958 | 162.007 |
| 11.000 | 3.436 | 3.437 | -36.132 | -25.388 |
| 12.000 | 3.052 | 4.356 | -23.486 | -20.141 |
| 13.000 | 2.798 | 4.759 | -20.143 | -18.862 |
| 14.000 | 2.732 | 4.534 | -19.639 | -17.785 |
| 15.000 | 2.858 | 3.800 | -21.386 | -16.544 |
| 16.000 | 3.127 | 2.843 | -26.446 | -14.461 |
| 17.000 | 3.465 | 1.977 | -52.327 | 44.052 |
| 18.000 | 3.786 | 1.408 | -27.372 | 162.220 |
| 19.000 | 4.021 | 1.226 | -22.550 | 164.595 |
| 20.000 | 4.127 | 1.409 | -21.123 | 166.195 |
| 21.000 | 4.091 | 1.875 | -21.842 | 167.810 |
| 22.000 | 3.925 | 2.497 | -24.973 | 169.936 |
| 23.000 | 3.656 | 3.123 | -33.629 | 175.639 |
| 24.000 | 3.349 | 3.597 | -34.860 | -17.874 |
| 25.000 | 3.066 | 3.783 | -25.827 | -11.242 |
| 26.000 | 2.869 | 3.619 | -22.823 | -8.811 |
| 27.000 | 2.802 | 3.152 | -22.190 | -6.789 |
| 28.000 | 2.876 | 2.534 | -23.449 | -4.626 |
| 29.000 | 3.065 | 1.956 | -27.165 | -1.577 |
| 30.000 | 3.312 | 1.574 | -37.803 | 9.430 |
| 31.000 | 3.568 | 1.469 | -33.920 | 169.660 |
| 32.000 | 3.782 | 1.640 | -26.432 | 176.327 |
| 33.000 | 3.917 | 2.025 | -23.665 | 179.261 |
| 34.000 | 3.953 | 2.533 | -22.956 | -178.282 |
| 35.000 | 3.886 | 3.053 | -23.878 | -175.732 |
| 36.000 | 3.725 | 3.478 | -26.730 | -172.471 |
| 37.000 | 3.491 | 3.708 | -33.525 | -165.437 |
| 38.000 | 3.228 | 3.670 | -42.176 | -18.272 |
| 39.000 | 2.977 | 3.340 | -29.614 | 3.649 |
| 40.000 | 2.780 | 2.760 | -25.507 | 8.068 |
| 41.000 | 2.670 | 2.038 | -23.814 | 11.125 |
| 42.000 | 2.660 | 1.332 | -23.586 | 13.933 |
| 43.000 | 2.734 | 0.796 | -24.668 | 16.926 |
| 44.000 | 2.878 | 0.539 | -27.353 | 20.705 |
| 45.000 | 3.063 | 0.605 | -33.075 | 27.920 |
| 46.000 | 3.257 | 0.977 | -47.750 | 128.511 |
| 47.000 | 3.427 | 1.581 | -32.425 | -166.778 |
| 48.000 | 3.553 | 2.339 | -27.395 | -160.205 |
| 49.000 | 3.614 | 3.149 | -25.019 | -156.375 |
| 50.000 | 3.601 | 3.921 | -24.018 | -153.154 |
| 51.000 | 3.522 | 4.561 | -24.019 | -150.050 |
| 52.000 | 3.213 | 5.002 | -29.394 | -121.519 |
| 53.000 | 3.033 | 5.483 | -29.406 | -101.627 |
| 54.000 | 2.835 | 5.620 | -29.416 | -82.155 |
| 55.000 | 2.629 | 5.394 | -29.422 | -63.056 |
| 56.000 | 2.432 | 4.817 | -29.425 | -44.351 |
| 57.000 | 2.271 | 3.936 | -29.428 | -26.068 |
| 58.000 | 2.160 | 2.848 | -29.430 | -8.191 |
| 59.000 | 2.106 | 1.676 | -29.424 | 9.275 |

TABLE 1 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 60.000 | 2.110 | 0.551 | -29.421 | 26.318 |
| 61.000 | 2.149 | -0.401 | -29.416 | 42.945 |
| 62.000 | 2.222 | -1.106 | -29.407 | 59.134 |
| 63.000 | 2.319 | -1.516 | -29.395 | 74.879 |
| 64.000 | 2.425 | -1.617 | -29.382 | 90.192 |
| 65.000 | 2.530 | -1.423 | -29.366 | 105.068 |
| 66.000 | 2.613 | -0.955 | -29.348 | 119.498 |
| 67.000 | 2.662 | -0.254 | -29.329 | 133.451 |
| 68.000 | 2.686 | 0.627 | -29.308 | 146.951 |
| 69.000 | 2.680 | 1.643 | -29.285 | 159.988 |
| 70.000 | 2.643 | 2.750 | -29.258 | 172.558 |
| 71.000 | 2.568 | 3.920 | -29.230 | -175.348 |
| 72.000 | 2.445 | 5.136 | -29.200 | -163.733 |
| 73.000 | 2.292 | 6.342 | -29.169 | -152.602 |
| 74.000 | 2.110 | 7.511 | -29.136 | -141.960 |
| 75.000 | 1.899 | 8.618 | -29.101 | -131.810 |
| 76.000 | 1.651 | 9.682 | -29.065 | -122.154 |
| 77.000 | 1.364 | 10.691 | -29.027 | -112.998 |
| 78.000 | 1.054 | 11.595 | -28.988 | -104.345 |
| 79.000 | 0.723 | 12.381 | -28.948 | -96.200 |
| 80.000 | 0.372 | 13.038 | -28.905 | -88.569 |
| 81.000 | -0.028 | 13.687 | -28.862 | -81.457 |
| 82.000 | -0.448 | 14.202 | -28.816 | -74.868 |
| 83.000 | -0.886 | 14.576 | -28.768 | -68.809 |
| 84.000 | -1.341 | 14.801 | -28.719 | -63.283 |
| 85.000 | -1.844 | 15.028 | -28.667 | -58.296 |
| 86.000 | -2.380 | 15.164 | -28.613 | -53.847 |
| 87.000 | -2.938 | 15.144 | -28.555 | -49.950 |
| 88.000 | -12.344 | 164.489 | -28.494 | -46.611 |
| 89.000 | -11.897 | 169.730 | -28.428 | -43.828 |
| 90.000 | -11.445 | 174.602 | -28.356 | -41.609 |
| 91.000 | -11.016 | -175.870 | -19.497 | 121.243 |
| 92.000 | -10.541 | -171.972 | -16.802 | 38.902 |
| 93.000 | -10.073 | -168.410 | -17.259 | -36.145 |
| 94.000 | -9.611 | -165.177 | -19.599 | -129.047 |
| 95.000 | -9.277 | 7.385 | -18.958 | 119.218 |
| 96.000 | -9.632 | 4.007 | -16.070 | 38.164 |
| 97.000 | -9.971 | 0.427 | -16.373 | -36.418 |
| 98.000 | -10.297 | -3.324 | -18.501 | -125.550 |
| 99.000 | -10.616 | -7.217 | -18.497 | 130.934 |
| 100.000 | -10.935 | -11.231 | -16.773 | 39.452 |
| 101.000 | -11.264 | -15.358 | -15.781 | -42.951 |
| 102.000 | -11.614 | -19.608 | -16.029 | -122.128 |
| 103.000 | -11.999 | -24.011 | -18.143 | 152.054 |
| 104.000 | -12.435 | -28.629 | -18.733 | 41.265 |
| 105.000 | -12.934 | -33.559 | -14.999 | -48.698 |
| 106.000 | -13.511 | -38.947 | -13.771 | -114.560 |
| 107.000 | -14.174 | -44.999 | -17.045 | -179.689 |
| 108.000 | -14.926 | -51.997 | -23.175 | 47.780 |
| 109.000 | -15.747 | -60.301 | -14.840 | -50.512 |
| 110.000 | -16.589 | -70.308 | -12.586 | -104.933 |
| 111.000 | -17.350 | -82.296 | -15.572 | -155.503 |
| 112.000 | -17.886 | -96.114 | -31.347 | 100.294 |
| 113.000 | -18.070 | -110.881 | -17.048 | -46.951 |
| 114.000 | -17.887 | -125.209 | -13.958 | -95.652 |
| 115.000 | -17.464 | -137.980 | -16.304 | -143.488 |
| 116.000 | -16.981 | -148.825 | -26.158 | 153.879 |
| 117.000 | -16.594 | -157.980 | -26.516 | -29.720 |
| 118.000 | -16.409 | -165.950 | -22.530 | -102.627 |
| 119.000 | -16.499 | -173.324 | -22.208 | 176.662 |
| 120.000 | -16.923 | 179.232 | -21.879 | 109.879 |
| 121.000 | -17.727 | 170.867 | -27.907 | 89.702 |
| 122.000 | -18.943 | 160.299 | -20.673 | 153.452 |
| 123.000 | -20.494 | 145.445 | -13.977 | 123.793 |
| 124.000 | -21.953 | 123.765 | -12.873 | 84.922 |
| 125.000 | -22.359 | 96.297 | -17.393 | 37.869 |

TABLE 1 - CONTINUED

| | | | |
|---|---|---|---|
| 126.000 | -21.360 | 71.195 | -23.943 | -125.896 |
| 127.000 | -19.886 | 53.391 | -13.125 | 161.027 |
| 128.000 | -18.710 | 41.218 | -10.367 | 107.371 |
| 129.000 | -18.097 | 32.077 | -10.012 | 45.000 |
| 130.000 | -18.138 | 24.074 | -9.434 | -23.032 |
| 131.000 | -18.908 | 15.572 | -9.109 | -83.997 |
| 132.000 | -20.516 | 4.273 | -11.301 | -144.182 |
| 133.000 | -22.958 | -14.596 | -15.040 | 119.867 |
| 134.000 | -25.067 | -49.050 | -10.014 | 26.587 |
| 135.000 | -24.005 | -89.187 | -6.675 | -26.338 |
| 136.000 | -21.406 | -113.440 | -6.420 | -73.276 |
| 137.000 | -19.516 | -126.700 | -8.311 | -125.962 |
| 138.000 | -18.664 | -135.565 | -10.587 | 170.781 |
| 139.000 | -18.865 | -143.285 | -12.463 | 111.840 |
| 140.000 | -20.237 | -152.437 | -18.950 | 69.393 |
| 141.000 | -23.058 | -167.949 | -19.962 | -162.909 |
| 142.000 | -26.790 | 155.230 | -10.157 | 166.680 |
| 143.000 | -25.593 | 99.698 | -7.625 | 130.548 |
| 144.000 | -21.774 | 72.879 | -8.386 | 86.059 |
| 145.000 | -19.455 | 60.856 | -10.291 | 21.468 |
| 146.000 | -18.648 | 53.193 | -9.322 | -50.572 |
| 147.000 | -19.273 | 46.076 | -8.298 | -107.620 |
| 148.000 | -21.603 | 35.912 | -9.148 | -171.024 |
| 149.000 | -26.327 | 10.905 | -7.247 | 109.496 |
| 150.000 | -28.144 | -60.304 | -2.835 | 53.305 |
| 151.000 | -22.661 | -98.779 | -0.060 | 12.541 |
| 152.000 | -19.454 | -111.230 | 1.071 | -26.026 |
| 153.000 | -18.290 | -118.061 | 1.348 | -68.206 |
| 154.000 | -18.873 | -124.054 | 1.832 | -113.652 |
| 155.000 | -21.564 | -132.927 | 2.787 | -155.895 |
| 156.000 | -27.820 | -160.568 | 3.396 | 167.843 |
| 157.000 | -27.697 | 107.218 | 2.998 | 134.388 |
| 158.000 | -21.131 | 79.684 | 1.428 | 98.215 |
| 159.000 | -18.196 | 71.397 | -0.686 | 52.274 |
| 160.000 | -17.502 | 66.474 | -1.413 | -2.440 |
| 161.000 | -18.863 | 61.436 | -0.771 | -51.378 |
| 162.000 | -23.280 | 51.187 | -0.501 | -96.343 |
| 163.000 | -32.740 | -22.193 | 0.036 | -147.008 |
| 164.000 | -22.751 | -93.849 | -0.261 | 168.168 |
| 165.000 | -17.927 | -103.490 | 0.340 | 128.896 |
| 166.000 | -16.203 | -107.588 | 1.176 | 100.487 |
| 167.000 | -16.688 | -110.831 | 0.989 | 78.534 |
| 168.000 | -19.926 | -115.819 | -0.656 | 56.283 |
| 169.000 | -30.526 | -143.949 | -3.636 | 22.930 |
| 170.000 | -23.418 | 87.685 | -4.605 | -31.198 |
| 171.000 | -16.655 | 78.274 | -1.747 | -70.944 |
| 172.000 | -14.057 | 75.350 | 0.335 | -90.787 |
| 173.000 | -13.824 | 73.536 | 0.602 | -103.521 |
| 174.000 | -12.167 | 75.921 | -2.290 | -114.182 |
| 175.000 | -12.749 | 75.249 | -9.559 | -148.402 |
| 176.000 | -14.669 | 74.782 | -6.047 | 109.295 |
| 177.000 | -18.225 | 74.437 | 1.039 | 91.209 |
| 178.000 | -24.278 | 74.069 | 4.524 | 87.007 |
| 179.000 | -35.629 | 72.724 | 6.287 | 85.284 |
| 180.000 | -126.056 | -140.389 | 6.886 | 84.371 |
| 0.000 | 0.967 | 8.395 | -123.560 | -71.142 |
| 1.000 | 1.210 | 7.958 | -108.371 | 44.911 |
| 2.000 | 1.861 | 6.773 | -108.842 | -97.689 |
| 3.000 | 2.718 | 5.288 | -119.681 | -2.060 |
| 4.000 | 3.577 | 3.892 | -126.287 | -23.311 |
| 5.000 | 4.281 | 2.773 | -133.494 | -69.934 |
| 6.000 | 4.739 | 1.983 | -126.804 | -44.951 |
| 7.000 | 4.653 | 3.521 | -134.996 | 11.195 |
| 8.000 | 4.618 | 3.677 | -134.950 | 2.367 |
| 9.000 | 4.333 | 3.636 | -135.487 | -6.151 |
| 10.000 | 3.876 | 3.377 | -136.588 | -14.116 |

36-16

TABLE 1 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 11.000 | 3.349 | 2.914 | -138.240 | -20.795 |
| 12.000 | 2.878 | 2.319 | -140.439 | -24.253 |
| 13.000 | 2.581 | 1.743 | -143.030 | -20.444 |
| 14.000 | 2.530 | 1.376 | -144.761 | -4.316 |
| 15.000 | 2.714 | 1.343 | -143.809 | 15.470 |
| 16.000 | 3.056 | 1.651 | -141.367 | 24.313 |
| 17.000 | 3.459 | 2.198 | -139.124 | 24.548 |
| 18.000 | 3.821 | 2.839 | -137.477 | 20.746 |
| 19.000 | 4.067 | 3.444 | -136.424 | 15.226 |
| 20.000 | 4.157 | 3.898 | -135.912 | 8.941 |
| 21.000 | 4.082 | 4.118 | -135.898 | 2.312 |
| 22.000 | 3.862 | 4.040 | -136.341 | -4.418 |
| 23.000 | 3.535 | 3.648 | -137.212 | -10.839 |
| 24.000 | 3.179 | 2.974 | -138.470 | -16.312 |
| 25.000 | 2.870 | 2.143 | -140.064 | -19.604 |
| 26.000 | 2.676 | 1.356 | -141.846 | -18.521 |
| 27.000 | 2.633 | 0.837 | -143.340 | -10.494 |
| 28.000 | 2.740 | 0.742 | -143.623 | 3.359 |
| 29.000 | 2.955 | 1.098 | -142.431 | 15.375 |
| 30.000 | 3.209 | 1.814 | -140.674 | 21.092 |
| 31.000 | 3.457 | 2.727 | -139.058 | 21.847 |
| 32.000 | 3.648 | 3.663 | -137.819 | 19.738 |
| 33.000 | 3.748 | 4.465 | -136.994 | 16.040 |
| 34.000 | 3.744 | 5.006 | -136.564 | 11.388 |
| 35.000 | 3.636 | 5.195 | -136.481 | 6.137 |
| 36.000 | 3.439 | 4.984 | -136.715 | 0.529 |
| 37.000 | 3.180 | 4.376 | -137.230 | -5.210 |
| 38.000 | 2.910 | 3.430 | -137.980 | -10.746 |
| 39.000 | 2.673 | 2.289 | -138.928 | -15.558 |
| 40.000 | 2.505 | 1.162 | -140.035 | -18.869 |
| 41.000 | 2.427 | 0.275 | -141.242 | -19.607 |
| 42.000 | 2.438 | -0.191 | -142.407 | -16.565 |
| 43.000 | 2.509 | -0.141 | -143.254 | -9.104 |
| 44.000 | 2.624 | 0.409 | -143.394 | 1.416 |
| 45.000 | 2.760 | 1.356 | -142.737 | 11.467 |
| 46.000 | 2.889 | 2.560 | -141.628 | 18.422 |
| 47.000 | 2.989 | 3.855 | -140.459 | 22.011 |
| 48.000 | 3.045 | 5.099 | -139.441 | 23.043 |
| 49.000 | 3.036 | 6.173 | -138.658 | 22.328 |
| 50.000 | 2.960 | 6.978 | -138.115 | 20.402 |
| 51.000 | 2.832 | 7.422 | -137.787 | 17.568 |
| 52.000 | 2.441 | 6.683 | -140.084 | 6.671 |
| 53.000 | 2.213 | 6.715 | -140.315 | 6.689 |
| 54.000 | 1.994 | 6.369 | -140.536 | 6.330 |
| 55.000 | 1.793 | 5.705 | -140.739 | 5.655 |
| 56.000 | 1.624 | 4.806 | -140.909 | 4.745 |
| 57.000 | 1.511 | 3.768 | -141.024 | 3.702 |
| 58.000 | 1.456 | 2.726 | -141.079 | 2.661 |
| 59.000 | 1.455 | 1.808 | -141.082 | 1.751 |
| 60.000 | 1.496 | 1.116 | -141.042 | 1.073 |
| 61.000 | 1.545 | 0.745 | -140.995 | 0.721 |
| 62.000 | 1.601 | 0.693 | -140.943 | 0.693 |
| 63.000 | 1.657 | 0.946 | -140.891 | 0.971 |
| 64.000 | 1.701 | 1.477 | -140.853 | 1.527 |
| 65.000 | 1.725 | 2.241 | -140.835 | 2.316 |
| 66.000 | 1.711 | 3.219 | -140.857 | 3.315 |
| 67.000 | 1.649 | 4.379 | -140.928 | 4.494 |
| 68.000 | 1.554 | 5.634 | -141.032 | 5.763 |
| 69.000 | 1.427 | 6.941 | -141.170 | 7.079 |
| 70.000 | 1.267 | 8.261 | -141.341 | 8.401 |
| 71.000 | 1.066 | 9.595 | -141.554 | 9.732 |
| 72.000 | 0.811 | 10.973 | -141.820 | 11.099 |
| 73.000 | 0.529 | 12.283 | -142.115 | 12.389 |
| 74.000 | 0.220 | 13.503 | -142.437 | 13.580 |
| 75.000 | -0.113 | 14.612 | -142.782 | 14.652 |
| 76.000 | -0.486 | 15.711 | -143.168 | 15.701 |

TABLE 1 - CONTINUED

| | | | |
|---|---|---|---|
| 77.000 | -0.899 | 16.802 | -143.593 | 16.732 |
| 78.000 | -1.330 | 17.761 | -144.037 | 17.618 |
| 79.000 | -1.778 | 18.581 | -144.495 | 18.352 |
| 80.000 | -2.240 | 19.253 | -144.967 | 18.925 |
| 81.000 | -2.763 | 20.180 | -145.498 | 19.738 |
| 82.000 | -3.302 | 20.986 | -146.044 | 20.413 |
| 83.000 | -3.856 | 21.659 | -146.603 | 20.938 |
| 84.000 | -4.426 | 22.192 | -147.174 | 21.306 |
| 85.000 | -5.052 | 23.127 | -147.800 | 22.052 |
| 86.000 | -5.717 | 24.163 | -148.461 | 22.874 |
| 87.000 | -6.410 | 25.098 | -149.146 | 23.563 |
| 88.000 | -7.137 | 25.929 | -149.859 | 24.113 |
| 89.000 | -7.934 | 27.678 | -150.640 | 25.524 |
| 90.000 | -8.797 | 29.780 | -151.482 | 27.216 |
| 91.000 | -6.894 | 0.484 | -148.344 | -16.725 |
| 92.000 | -18.200 | -11.106 | -148.576 | -16.249 |
| 93.000 | -10.874 | 77.345 | -148.892 | -16.462 |
| 94.000 | -7.008 | 41.726 | -149.657 | -16.291 |
| 95.000 | -9.582 | -4.731 | -149.645 | -8.230 |
| 96.000 | -23.772 | -81.953 | -148.810 | -5.792 |
| 97.000 | -11.906 | 93.435 | -148.414 | -8.293 |
| 98.000 | -7.499 | 39.802 | -149.023 | -7.558 |
| 99.000 | -8.685 | -13.869 | -149.105 | 1.380 |
| 100.000 | -17.197 | -89.486 | -148.004 | 5.063 |
| 101.000 | -13.145 | 102.643 | -147.904 | 1.059 |
| 102.000 | -7.699 | 39.587 | -149.391 | 3.439 |
| 103.000 | -7.980 | -17.712 | -149.688 | 18.939 |
| 104.000 | -13.526 | -92.381 | -148.304 | 22.707 |
| 105.000 | -13.795 | 124.896 | -148.867 | 16.127 |
| 106.000 | -8.428 | 47.382 | -152.028 | 22.285 |
| 107.000 | -8.046 | -16.087 | -152.133 | 51.304 |
| 108.000 | -11.181 | -92.745 | -150.251 | 52.323 |
| 109.000 | -12.637 | 158.138 | -152.425 | 42.005 |
| 110.000 | -10.031 | 68.311 | -159.654 | 73.363 |
| 111.000 | -9.498 | -7.565 | -154.119 | 119.383 |
| 112.000 | -10.202 | -88.751 | -152.643 | 108.530 |
| 113.000 | -10.592 | -171.804 | -158.388 | 113.013 |
| 114.000 | -11.775 | 106.971 | -155.670 | -172.593 |
| 115.000 | -13.215 | 12.420 | -150.573 | -177.906 |
| 116.000 | -11.403 | -78.644 | -151.225 | 172.105 |
| 117.000 | -10.477 | -146.432 | -154.686 | -165.454 |
| 118.000 | -13.337 | 150.701 | -151.474 | -134.857 |
| 119.000 | -20.425 | 48.160 | -149.593 | -139.022 |
| 120.000 | -16.017 | -66.398 | -151.620 | -141.711 |
| 121.000 | -15.061 | -125.226 | -154.447 | -116.226 |
| 122.000 | -22.027 | 176.509 | -153.209 | -96.305 |
| 123.000 | -23.835 | -14.532 | -154.992 | -93.396 |
| 124.000 | -17.874 | -80.593 | -159.169 | -48.148 |
| 125.000 | -19.409 | -168.018 | -154.261 | -18.339 |
| 126.000 | -14.850 | 84.520 | -154.579 | -24.324 |
| 127.000 | -10.714 | 17.928 | -162.049 | 28.088 |
| 128.000 | -10.145 | -47.197 | -151.447 | 69.991 |
| 129.000 | -10.061 | -129.731 | -147.878 | 50.578 |
| 130.000 | -7.329 | 151.685 | -149.828 | 23.129 |
| 131.000 | -5.465 | 88.954 | -164.406 | -38.737 |
| 132.000 | -5.620 | 25.468 | -151.496 | 161.625 |
| 133.000 | -5.886 | -50.035 | -147.536 | 125.276 |
| 134.000 | -4.199 | -123.032 | -149.294 | 80.832 |
| 135.000 | -2.897 | 176.848 | -152.505 | -9.266 |
| 136.000 | -3.216 | 118.732 | -147.352 | -85.919 |
| 137.000 | -4.208 | 52.928 | -145.211 | -127.239 |
| 138.000 | -4.089 | -15.320 | -147.188 | -164.968 |
| 139.000 | -3.911 | -74.246 | -154.608 | 133.344 |
| 140.000 | -5.456 | -130.236 | -152.638 | 12.120 |
| 141.000 | -8.351 | 159.964 | -149.009 | -35.315 |
| 142.000 | -8.361 | 78.855 | -151.072 | -76.168 |

36-18

TABLE 1 - CONTINUED

| | | | | |
|---|---|---|---|---|
| 143.000 | -7.437 | 18.387 | -157.705 | -160.880 |
| 144.000 | -9.546 | -36.077 | -152.278 | 104.122 |
| 145.000 | -14.118 | -123.312 | -150.306 | 63.331 |
| 146.000 | -10.246 | 139.402 | -154.792 | 21.759 |
| 147.000 | -7.191 | 79.624 | -159.212 | -130.381 |
| 148.000 | -5.950 | 16.532 | -148.652 | 173.045 |
| 149.000 | -3.189 | -49.164 | -145.300 | 132.897 |
| 150.000 | 0.210 | -100.484 | -143.992 | 85.835 |
| 151.000 | 2.431 | -142.558 | -142.335 | 33.572 |
| 152.000 | 3.450 | 176.693 | -140.091 | -12.930 |
| 153.000 | 3.908 | 133.344 | -138.407 | -52.429 |
| 154.000 | 4.568 | 88.690 | -137.622 | -89.613 |
| 155.000 | 5.464 | 47.481 | -137.447 | -127.713 |
| 156.000 | 5.985 | 11.041 | -137.452 | -166.868 |
| 157.000 | 5.714 | -23.198 | -137.460 | 154.887 |
| 158.000 | 4.637 | -58.896 | -137.749 | 118.969 |
| 159.000 | 3.189 | -98.981 | -138.500 | 85.279 |
| 160.000 | 2.029 | -142.197 | -140.106 | 49.791 |
| 161.000 | 0.969 | 176.041 | -142.455 | 10.842 |
| 162.000 | -1.089 | 132.861 | -145.656 | -36.659 |
| 163.000 | -3.617 | 68.768 | -148.418 | -108.961 |
| 164.000 | -3.954 | 14.578 | -143.697 | -157.234 |
| 165.000 | -2.785 | -25.604 | -144.035 | 166.474 |
| 166.000 | -1.611 | -52.040 | -143.844 | 134.127 |
| 167.000 | -1.739 | -71.742 | -143.912 | 106.757 |
| 168.000 | -3.557 | -92.097 | -146.741 | 81.996 |
| 169.000 | -7.069 | -125.711 | -148.707 | 51.470 |
| 170.000 | -7.892 | 173.963 | -149.858 | 7.639 |
| 171.000 | -4.239 | 135.206 | -148.471 | -33.268 |
| 172.000 | -1.952 | 117.796 | -146.925 | -59.668 |
| 173.000 | -1.660 | 106.856 | -146.352 | -80.852 |
| 174.000 | -3.881 | 93.067 | -125.159 | -119.108 |
| 175.000 | -15.556 | 119.334 | -126.578 | -89.548 |
| 176.000 | -5.983 | -107.781 | -123.165 | -74.731 |
| 177.000 | 1.144 | -100.359 | -124.399 | -129.605 |
| 178.000 | 4.542 | -97.221 | -102.520 | 111.021 |
| 179.000 | 6.281 | -95.619 | -94.737 | -10.519 |
| 180.000 | 6.886 | -95.629 | -101.403 | 27.062 |

```
TO: LAIC=F, LGTD=F, LFEED=T
F    0.   0.
F    0    0    0    0
F    F    F    F    0
F    T    F    0.8
F    F    F    T    F    F    0.   0.
F    F    0.   0.
PG:
4    1
2
12.05        -3.0        3.5
12.05        -3.0        7.5
12.05         3.0        7.5
12.05         3.0        3.5
TP:
F    0
T    T    T    T    F
0.8
F    T    F    F
PZ:
1
0.
0.  180.  1
F
PP:
1
1    1
1    2
XQ:
```

The calculated H-plane pattern of the main reflector without shroud is given in Figure 7. The pattern from the flat plate simulation is shown in Figure 8; the envelope of the measured pattern with shroud is also included. The measured pattern is shown in Figure 9. Note that the contribution from the aperture fields of the reflector are not included in the plate simulation. However, the contributions from the feed spillover and the shroud edge dominate the pattern for angles beyond 30° as can be seen from the calculated patterns in Figures 7 and 8. From this simulation, one can see the effects of the shroud on the spillover fields of the reflector.
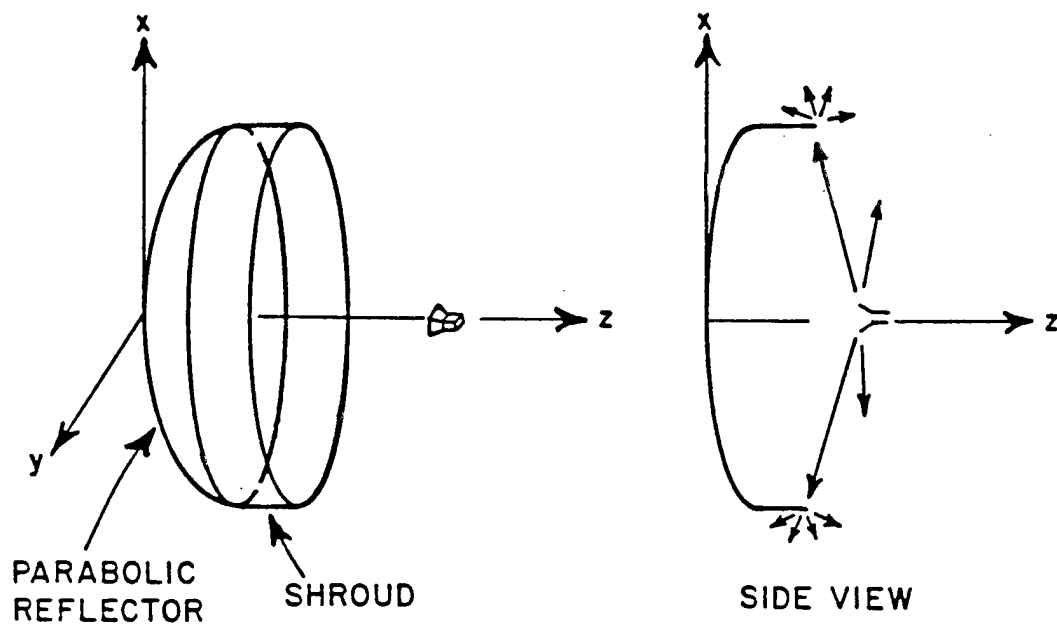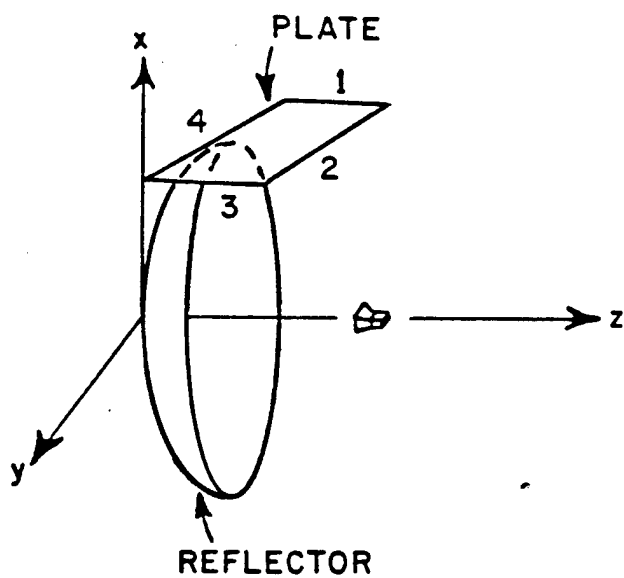
Figure 5. Geometry of reflector with shroud.



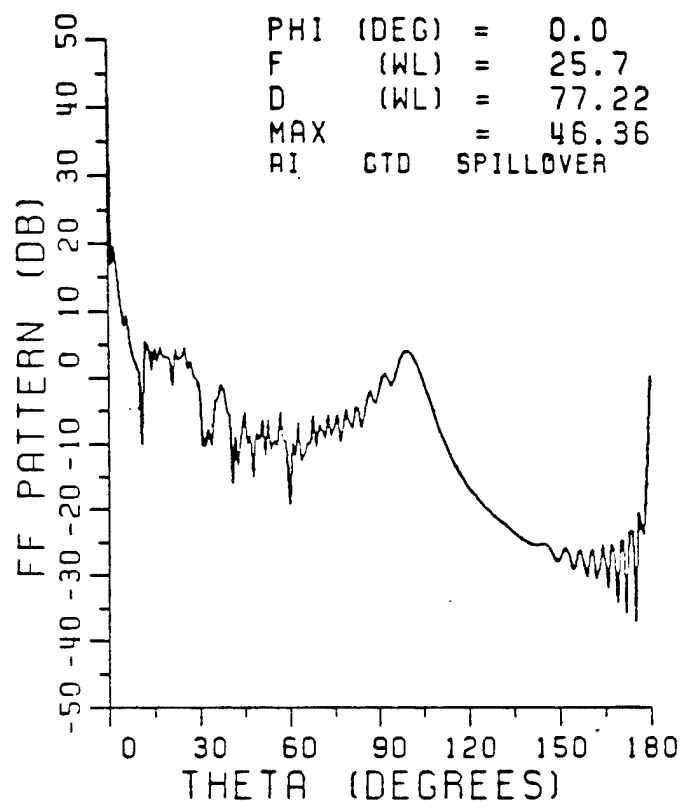Figure 6. Geometry of reflector with one plate.

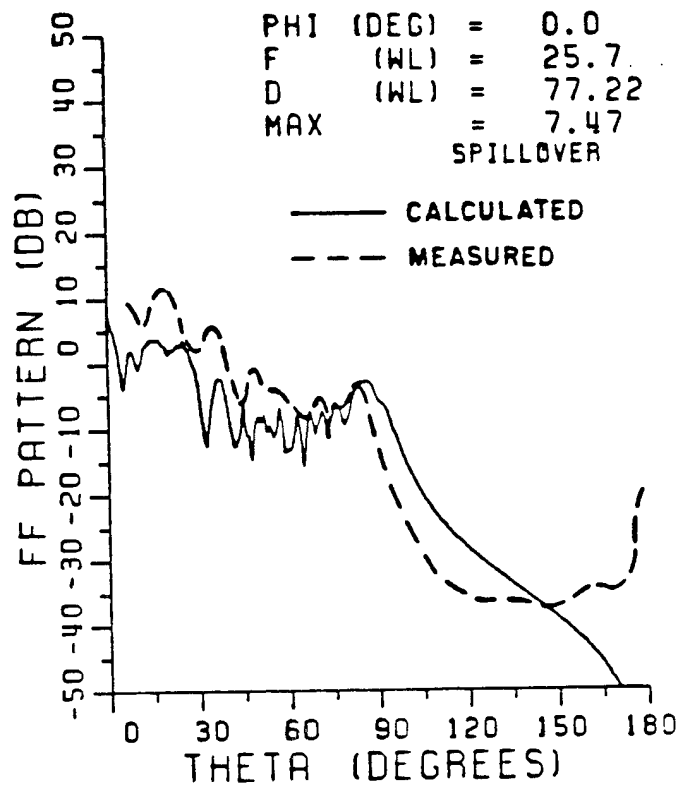Figure 7. Calculated H-plane pattern of the reflector without plate.

Figure 8. Calculated (solid line) pattern of flat plate simulation with
comparison to the measured pattern of the reflector with
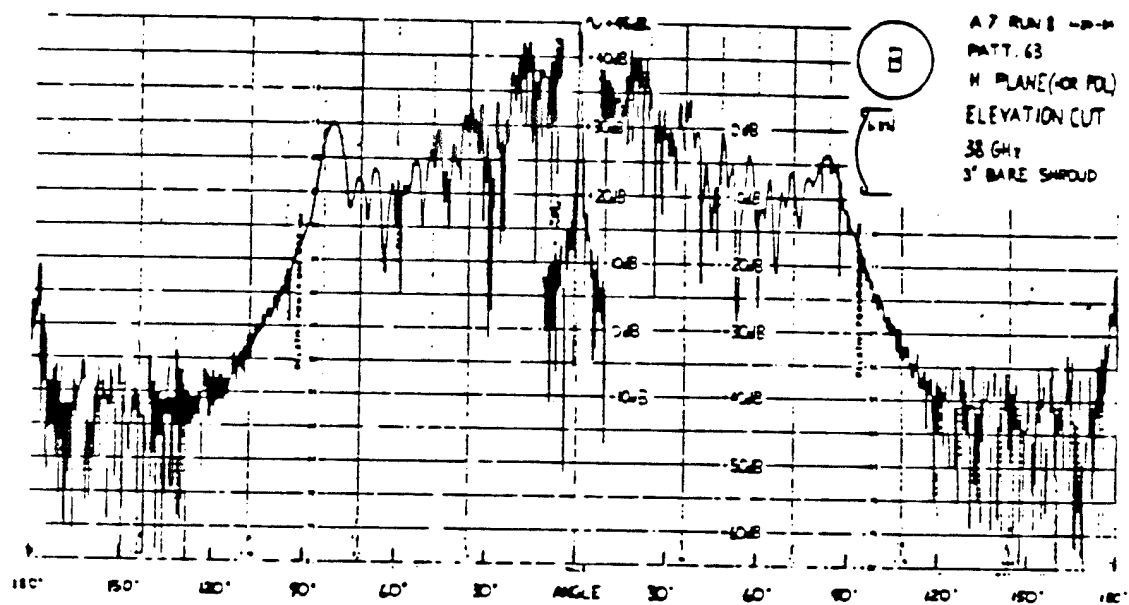shroud. (dashed line)

Figure 9. Measured H-plane pattern of the reflector with shroud.

---------------------------------------------------------------
Command RT: ROTATE/TRANSLATE
---------------------------------------------------------------

This command enables the user to translate and/or rotate the coordinate system used to define the input data in order to simplify input data in the PG: Command for the geometries of plate scatterers or to simplify input data in the CL: Command for linear antenna position used for coupling. The data in any PG: or CL: Command that follows RT: will be transformed. This command affects only the PG: and CL: Commands. The geometry is illustrated in Figure 1.

1. Read: (TR(N),N=1,3)

   a) TR(N): This dimensioned real variable (in input units) is used to specify the origin of the new coordinate system (referred to the reflector vertex) to be used to input the data for the plates or coupling.

2. Read: THZP,PHZP,THXP,PHXP

   a) THZP,PHZP: These real variables are input in degrees as spherical angles that define the z-axis of the new coordinate system as if it was a radial vector in the reflector coordinate system.

   b) THXP,PHXP: These real variables are input in degrees as spherical angles that define the x-axis of the new coordinate system as if it was a radial vector in the reflector coordinate system.

The new x-axis and z-axis must be defined orthogonal to each other. The new y-axis is found from the cross product of the x and z-axis. All the subsequent inputs for the PG: and CL: Commands will be made relative to this new coordinate system, which is shown as xt, yt, zt, unless command "RT:" is called again and redefined.
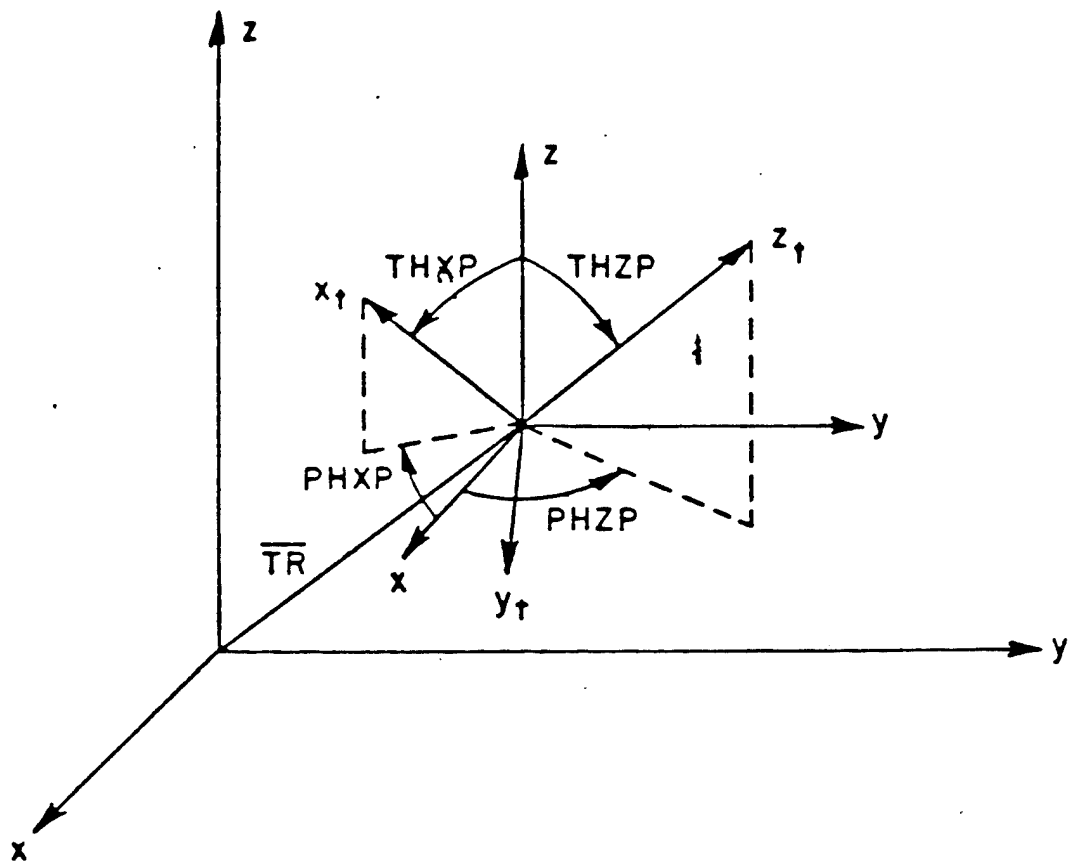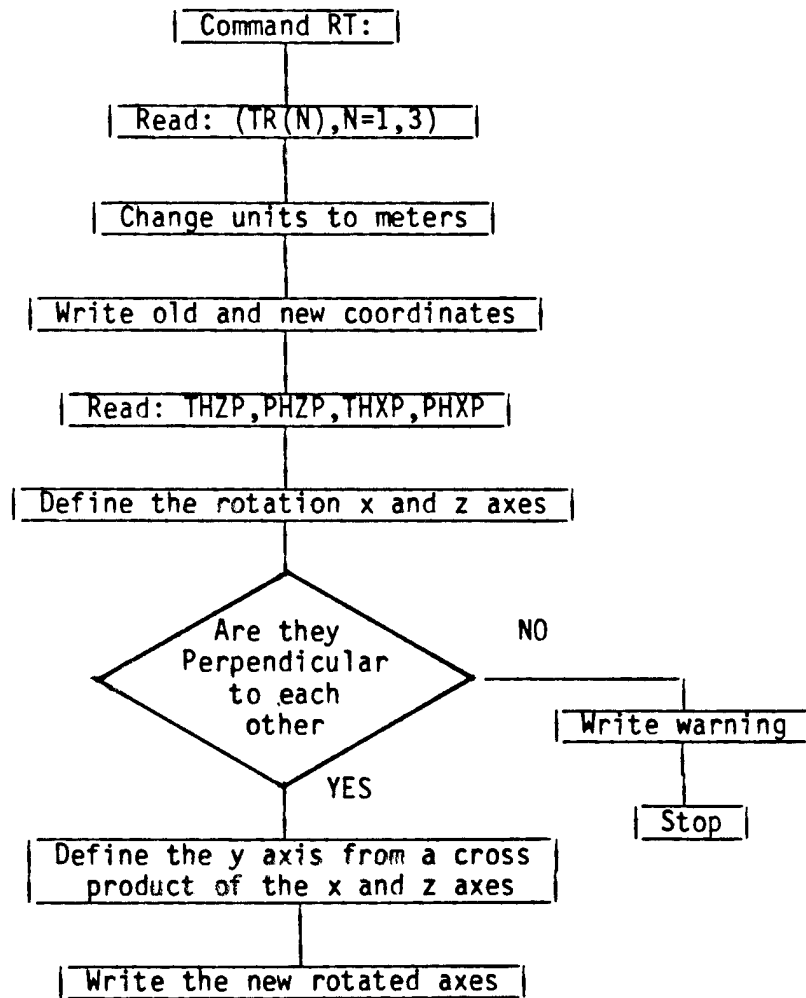
Figure 1. Definition of rotate-translate coordinate system geometry.

# Block Diagram for Rotation and Translation

```
                    | Command RT: |
                          |
              | Read: (TR(N),N=1,3) |
                          |
              | Change units to meters |
                          |
          | Write old and new coordinates |
                          |
          | Read: THZP,PHZP,THXP,PHXP |
                          |
        | Define the rotation x and z axes |
                          |
                         /\
                        /  \
                       / Are they \
                      / Perpendicular \          NO
                      \  to each      / ----------------
                       \  other      /                 |
                        \           /          | Write warning |
                         \/                            |
                         YES                        | Stop |
                          |
        | Define the y axis from a cross |
        | product of the x and z axes |
                          |
          | Write the new rotated axes |
```

37-3

```
----------------------------------------------------------------
```
## Command CL:   REFLECTOR TO LINEAR COUPLING
```
----------------------------------------------------------------
```

This command enables the user to define a linear antenna for which the coupling to the reflector antenna can be calculated.

NOTE 1:   The results from the computer code assume that both antennas are perfectly matched. The computed coupling result should be multiplied by the loss factor (mismatch in power and antenna efficiency loss) for both the receiver and transmitter.

NOTE 2:   By reciprocity, the coupling between the reflector antenna and a linear antenna is the same when the transmitter and receiver are interchanged, at the same frequency. However, it should be noted that a linear antenna will typically transmit at a frequency below the cutoff frequency of the reflector feed system. Thus, the impedance mismatch will usually cause very low coupling from linear to reflector. The linear antenna will typically receive the transmitted signal from the reflector antenna at a higher coupling level because of less impedance mismatch.

1.  Read:  NLA

    a) NLA:  This integer variable is used to define the number of points on the linear antenna at which the incident field from the reflector is to be calculated for reaction with the currents on the linear antenna. If NLA=1 the incident field of the reflector is calculated only at the midpoint of the linear antenna. Presently 1<NLA<901.

2.  Read:  (XLA1(N),N=1,3)

    a) XLA1(N):  This dimensioned real variable defines one end-point of the linear antenna.

3.  Read:  (XLA2(N),N=1,3)

    a) XLA2(N):  This dimensioned real variable defines the other end-point of the linear antenna which should be different from XLA1(N), even for NLA=1.

4. Read:  NPN,LWCLA

   a) NPN:  This integer variable is used to specify the number of
      linear antenna locations desired.  Presently 1<NPN<1842.

   b) LWCLA:  This logical variable, if set true, is used to
      output the end-point coordinates of the linear antenna at
      all NPN locations.  If set false, only the data for the
      initial location is output.  (normally set false)

5. Read:  LRECT

   This statement is skipped if NPN=1.

   a) LRECT:  This logical variable, if set true, specifies the
      linear antenna to move along rectangular coordinates.  If
      set false, the reflector antenna is rotated (LROT=true) or
      tilted (LROT=false); so that the linear antenna moves along
      spherical coordinates relative to the reflector antenna.

      LRECT=TRUE     TRANSLATE LINEAR ANTENNA ALONG RECTANGULAR
                     COORDINATES
      LRECT=FALSE    ROTATE OR TILT REFLECTOR ANTENNA

6. Read:  (XRECI(N),N=1,3)

   This statement is skipped if NPN=1 or LRECT=false.

   a) XRECI(N):  This dimensioned real variable defines the
      increment vector for LRECT=true.

7. Read:  LROT,(XRTC(N),N=1,3),ANGI

   This statement is skipped if NPN=1 or LRECT=true.

   a) LROT:  This logical variable is used to specify rotation or
      tilt of the reflector.  If set true, the reflector is
      rotated.  If set false, the reflector is tilted.

      LROT=TRUE      ROTATE REFLECTOR ANTENNA
      LROT=FALSE     TILT REFLECTOR ANTENNA

   b) XRTC(N):  This dimensioned real variable defines the center
      of rotation or tilt for the reflector (referred to the
      reflector vertex).

   c) ANGI:  This real variable defines the increment in degrees
      for rotation or tilt of the reflector.

8. Read: LINCON

   a) LINCON: This logical variable, if set true, provides for
      the calculation of the incident field from the reflector
      antenna (at NLA points between XLA1 and XLA2) without the
      calculation of coupling. If set false, coupling is
      calculated. (normally set false)

      LINCON=TRUE   CALCULATE INCIDENT FIELD FROM THE REFLECTOR
                    ANTENNA
      LINCON=FALSE  CALCULATE COUPLING


9. Read: WMR(NN),WPR(NN)

   This statement is skipped if LINCON=true. This read statement
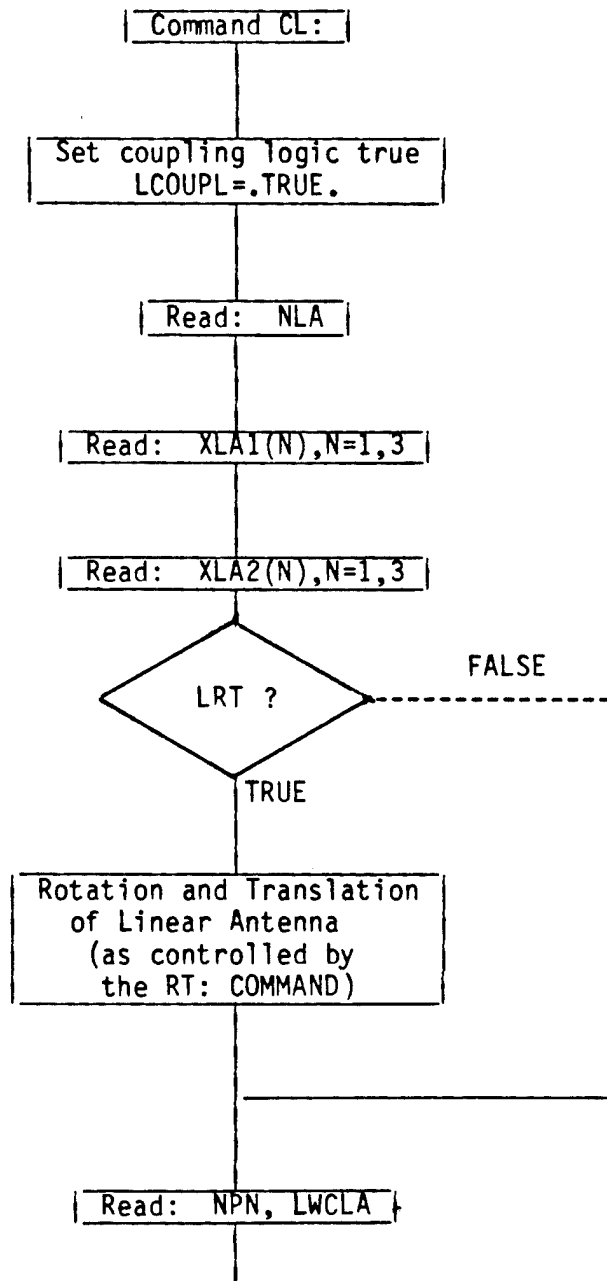   is executed NLA times.

   a) WMR(NN): This dimensioned real variable defines the
      magnitude of the current distribution on the linear antenna,
      at NLA uniformly spaced points.

   b) WPR(NN): This dimensioned real variable defines the phase
      of the current distribution on the linear antenna, at NLA
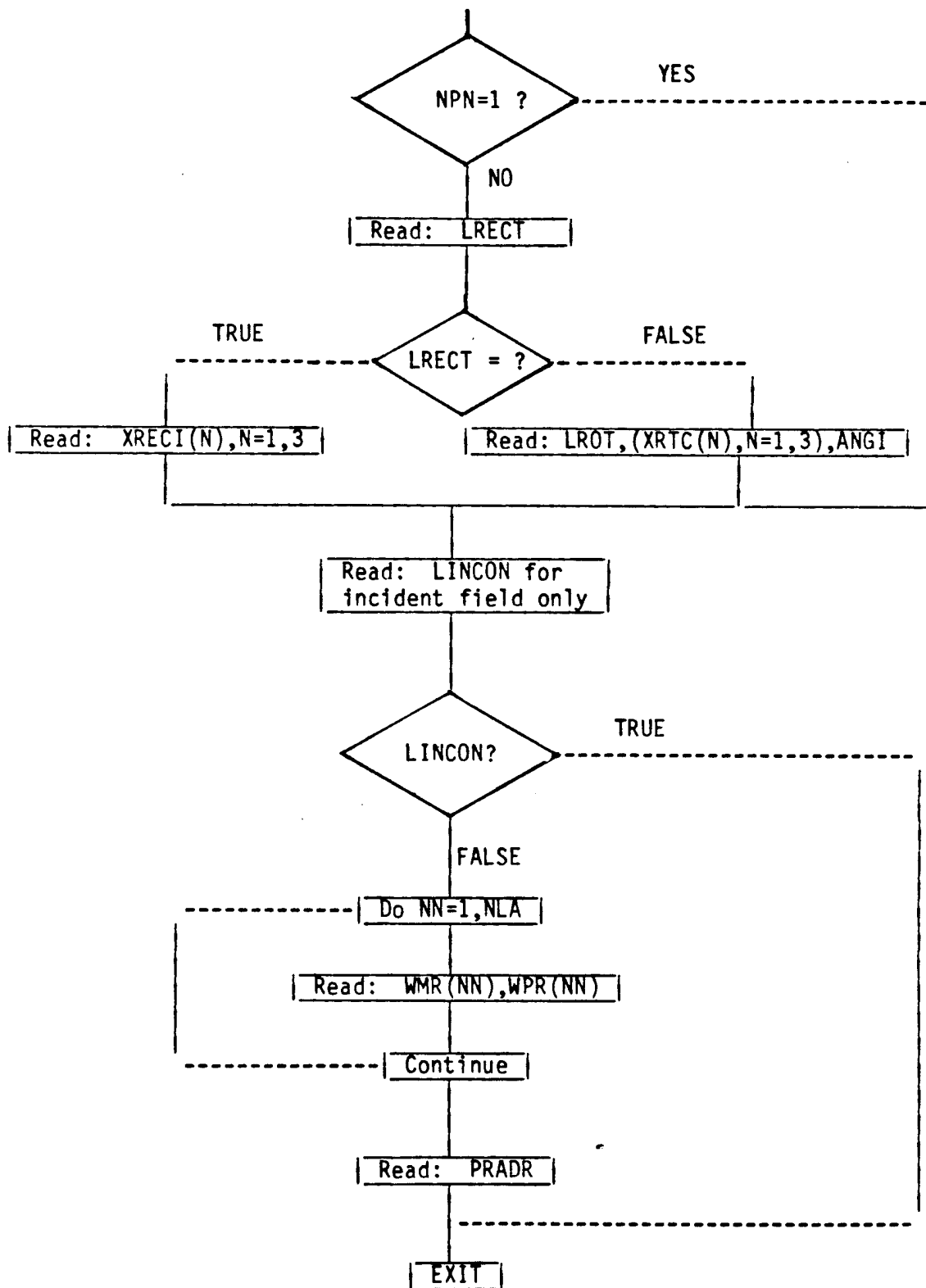      uniformly spaced points.


10. Read: PRADR

    This statement is skipped if LINCON=true.

    a) PRADR: This real variable specifies the radiated power from
       the linear antenna when it transmits with the currents
       specified for coupling calculations.

## FLOW DIAGRAM

| Command CL: |

| Set coupling logic true |
| LCOUPL=.TRUE. |

| Read:  NLA |

| Read:  XLA1(N),N=1,3 |

| Read:  XLA2(N),N=1,3 |

LRT ?

FALSE

TRUE

| Rotation and Translation |
| of Linear Antenna |
| (as controlled by |
| the RT: COMMAND) |

| Read:  NPN, LWCLA |

```
                         |
                        / \
                       /   \
                      /     \               YES
                     < NPN=1 ? >-----------------------------------------.
                      \     /                                            |
                       \   /                                             |
                        \ /                                              |
                         | NO                                            |
                         |                                               |
                 |  Read:  LRECT  |                                      |
                         |                                               |
                         |                                               |
        TRUE            / \                  FALSE                       |
     .--------------- < LRECT = ? >-----------------.                    |
     |                  \ /                          |                   |
     |                   |                           |                   |
     |                                               |                   |
| Read:  XRECI(N),N=1,3 |      | Read: LROT,(XRTC(N),N=1,3),ANGI |       |
     |                                               |                   |
     '-----------------------.--------------------------------------------'
                             |
                             |
                    | Read:  LINCON for |
                    | incident field only |
                             |
                             |
                            / \                      TRUE
                           /   \
                          <LINCON?>-------------------------------------.
                           \   /                                        |
                            \ /                                         |
                             | FALSE                                    |
                             |                                          |
        .------------------| Do NN=1,NLA |                              |
        |                    |                                          |
        |                    |                                          |
        |           | Read:  WMR(NN),WPR(NN) |                          |
        |                    |                                          |
        |                    |                                          |
        '------------------| Continue |                                |
                             |                                          |
                             |                                          |
                    | Read:  PRADR |                                    |
                             |                                          |
                             '------------------------------------------'
                             |
                        | EXIT |
```

---
Command NR:   NRUN OPTION (STORE YSUM DATA FOR NEXT RUN)
---

This command enables the user to store the YSUM data when the aperture integration is used. The stored YSUM data can be used for the next run when the parameters of the reflector system such as grid size, frequency and diameter are not changed. This option is useful since it saves the CPU time in calculating the aperture fields and the YSUM data, especially when strut scattering, feed blockage and plate scattering are included in the pattern calculation. One can input different geometries of struts and plates to obtain the total pattern without recalculating the aperture fields and YSUM data. Note that only one $\phi$-cut is allowed when this command is used since the YSUM data are different for different $\phi$-cuts.

1. Read:   NRUN

    a) NRUN:  This integer variable specifies whether the aperture index of the integration data is to be stored for future use, or read from a previous output file.

    NRUN = 1:  The YSUM data are stored in unit #21.

    NRUN = 2:  The YSUM data are read from unit #21.

BLOCK DIAGRAM FOR NRUN OPTION

```
| Command:   NR |
        |
        |
        |
| Read:   NRUN |
```
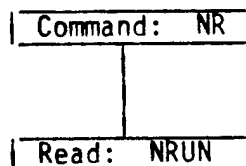
---
Command AS:  SPREAD FACTOR CALCULATION FOR G.O. FIELD
---


    This command enables the user to specify whether the spread factor
calculation for the Geometrical Optics (G.O.) field is used or not.
For focused parabolic reflectors, the spread factor for the G.O. field
is unity so that this command is not necessary.  For defocused parabolic
reflectors or other reflectors such as hyperbolic or elliptic
reflectors, the spread factor is not unity so that this command can be
used to calculate the spread factor.

    A numerical method or an analytic method can be used to calculate
the caustic distance of the reflected field and its associated spread
factor.  The choice is specified by the logical variable LSPFN.  With
the numerical method, two points adjacent to the reflection point are
used to calculate the caustic distance of the reflected field for each
of the RHO- and PHI-directions.  With the other option, an analytic
expression is used for the caustic distance.


BLOCK DIAGRAM FOR SPREAD FACTOR CALCULATION



40-1
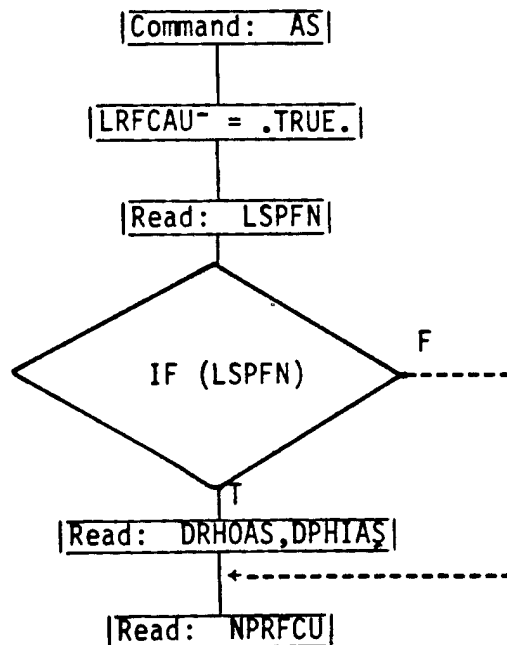
1. Read: LSPFN

    a) This logical variable specifies whether the numerical or the analytic method is used to calculate the reflected field caustic distance. If LSPFN=T, the numerical method is used. If LSPFN=F, the analytic expression is used.

2. READ: DRHOAS, DPHIAS

    This read statement is used only for LSPFN = T.

    a) DRHOAS: This variable specifies the distance (in wavelengths) from the reflection point to the two adjacent points for the RHO-direction.

    b) DPHIAS: This variable specifies the angle (in degrees) from the reflection point to the two adjacent points for the PHI-direction.

3. READ: NPRFCU

    a) NPRFCU: This variable specifies that the spread factor will be re-calculated for every NPRFCU field point. For example, if NPRFCU = 4, the spread factor is calculated for every 4th point; and the same value is used for the following 3 points.

---
Command RF:   REFLECTOR ROTATION AND TILT
---

This command enables the user to rotate and/or tilt the reflector antenna without changing the input of the near field points, plate or coupling geometry, so that the specification of the geometry is simplified.  It is useful when the near field ponits, the plate scatterer or the linear antenna (for coupling) is fixed and reflector needs to be scanned.  The geometry is illustrated in Figure 1.

1.  Read:  LRFRL,LRFRT,LRFTL

   a) LRFRL:  This is a logical variable specified by T or F.  If set true, the reflector may be rotated about Z-axis.

   b) LRFRT:  This is a logical variable specified by T or F.  If set true, the reflector may be rotated in azimuth.

   c) LRFTL:  This is a logical variable specified by T or F.  If set true, the reflector may be tilted in elevation.

2.  Read:  ANGRL

   a) ANGRL:  This real variable defines the angle of rotation about the z-axis with respect to the x-axis.

3.  Read:  XORT(I),I=1,3,ANGRT

   This statement is skipped if LRFRT is false.

   a) XORT:  This dimensioned real variable defines the coordinate of the rotation center.

   b) ANGRT:  This real variable defines the rotation angle in degrees.

4.  Read:  XOTL(I),I,ANGTL

   This statement is skipped if LRFTL is false.

   a) XOTL:  This dimensional real variable defines the coordinate of the tilt center.

   b) ANGTL:  This real variable defines the tilt angle in degrees.

```
                    | Command:  RF |
                           |
                           |
        | Read:  LRFRL,LRFRT,LRFTL |
                           |
                           |

                          /\
                         /  \                  FALSE
                        /    \      - - - - - - - - - -
                        \    /
                    \ LRFRL=? /
                        \    /
                         \  /
                          \/
                           |
                           |
                         TRUE

                    | Read:  ANGRL |
                           |
                           |

                          /\
                         /  \                  FALSE
                        /    \      - - - - - - - - - -
                        \    /
                    \ LRFRT=? /
                        \    /
                         \  /
                          \/
                           |
                           |
                         TRUE

                | Read:  XORT,ANGRT |
                           |
                           |

                          /\
                         /  \                  FALSE
                        /    \      - - - - - - - - - -
                        \    /
                    \ LRFTL=? /
                        \    /
                         \  /
                          \/
                           |
                           |
                         TRUE

                | Read:  XOTL,ANGTL |
                           |
                           |
                           |
```
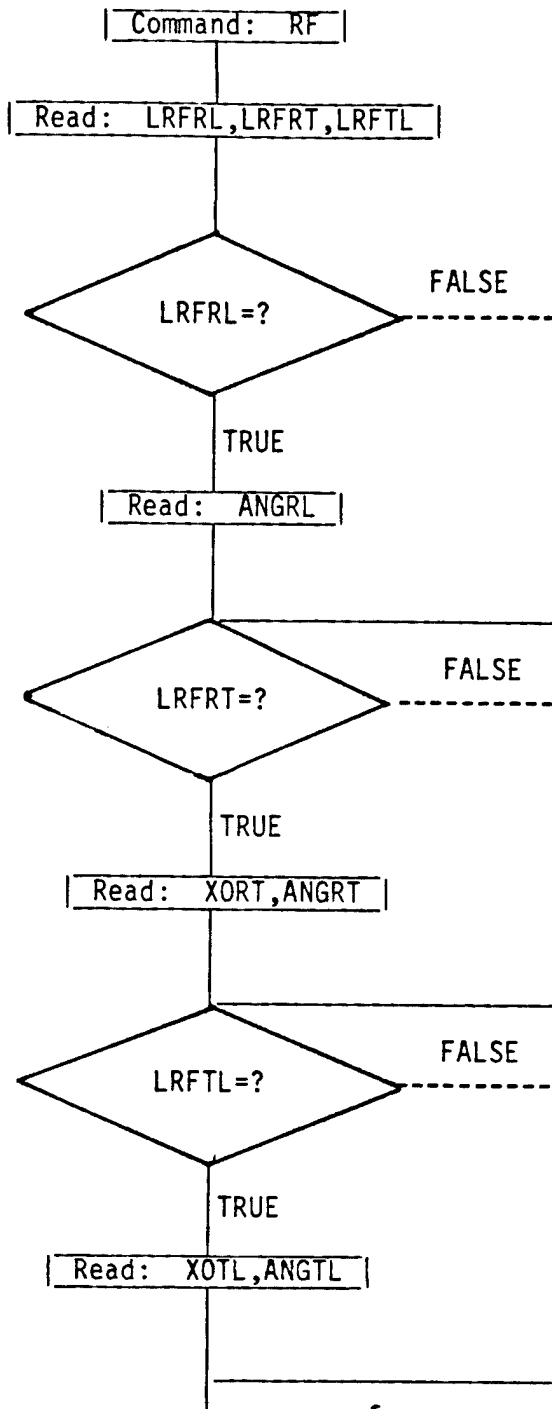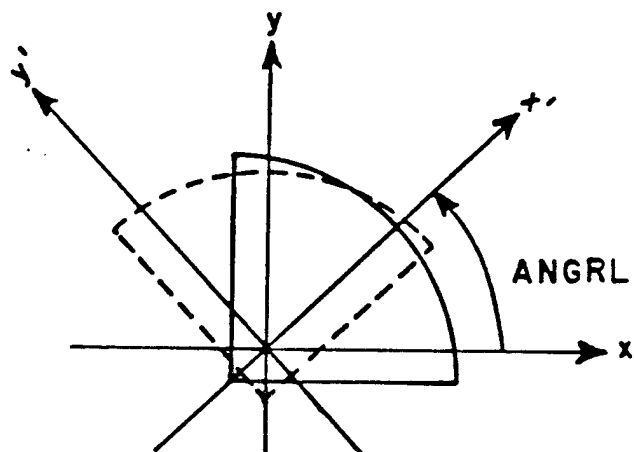
(a) Rotation about z-axis

Figure 1. Geometry for the rotation.
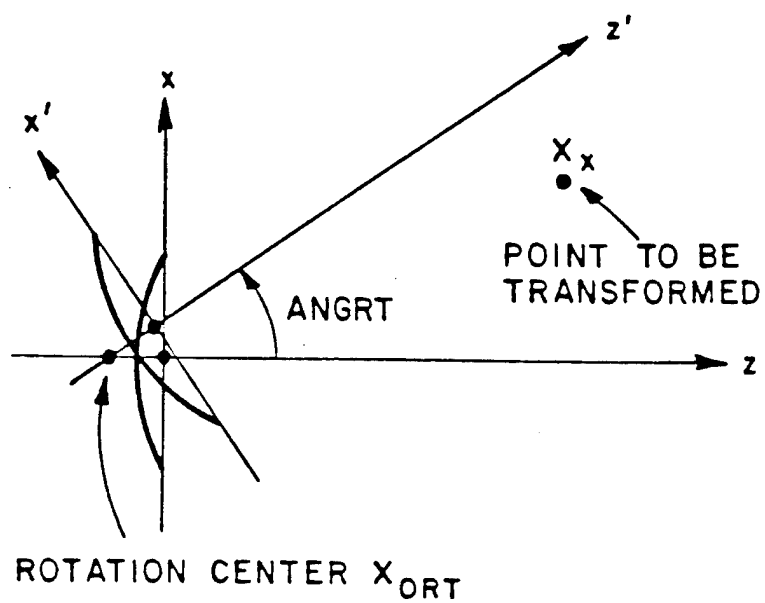
(b) Azimuth rotation
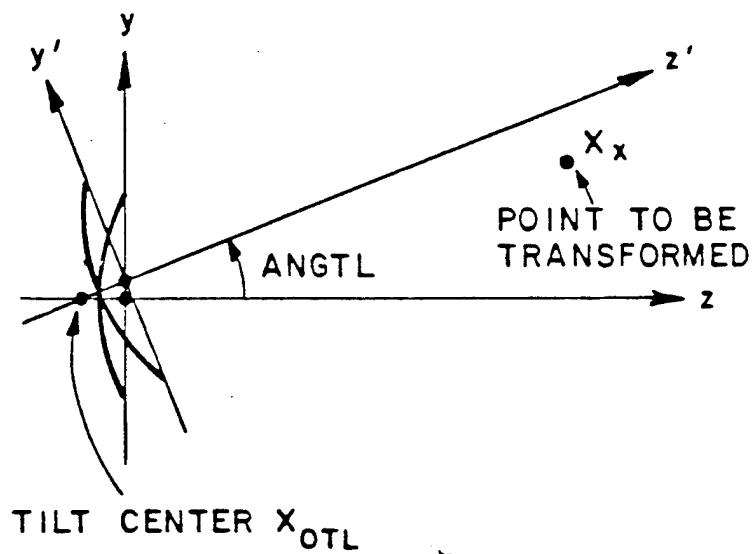


(c) Elevation tilt

Figure 1. Continued.

```
---------------------------------------------------------------
Command CK:   CRACK SCATTERING
---------------------------------------------------------------
```

This command specifies the geometry of thin cracks in the reflector surface. Each crack is assumed rectangular in shape and can be divided into small sections. Each section will be considered as an individual crack inside the code and can be further subdivided into small segments.

The axis of each crack is straight; however, a piecewise linear crack can be modeled by joining several cracks at their end-points. In such a case, the coordinates of the end point at each junction will be input as end-points for both adjacent cracks.

1.  Read:   NCK,GRCK,THECK,LWSAME,LNGSAM

    a) NCK:  This integer variable specifies the number of cracks. Presently, NCK<108.

    b) GRCK:  This real variable specifies the size of the segment used in subdividing a section of each crack. The grid size GRCK (input in units) controls the number of segments into which each section is subdivided, and should not exceed 0.5 wavelengths. Presently, a maximum of 63 segments can be used on any crack section.

    c) THECK:  This real variable is used to adjust the maximum theta angle for which crack scattering is included.

    d) LWSAME:  This logical variable specifies whether all the cracks have same width or not. If LWSAME is true, the crack width is input only once.

    e) LNGSAM:  This logical variable specifies whether all cracks are divided into the same number of sections or not. (Note that each section can be further subdivided into small segments by the variable GRCK).

2.  Read:  RHCKU,PHICK(M,J)

    This statement is executed twice for the end-pionts (J=1,2) for each crack (strut #M). The coordinate system is cylindrical, referred to the vertex of the reflector as shown in Figure 1.

    a) RHCKU:  This real variable specifies the rho-coordiante of the $j^{th}$ end point on the $M^{th}$ crack.

b) PHICK(M,J):  This is a real variable specifying the phi-coordinate (in degrees) of the end-points (J=1,2) on the M$^{th}$ crack.

3. Read:  NCC(M)

a) NCC(M):  This integer variable specifies the number of sections into which crack M is to be subdivided.  (This is useful for cracks which are too long to be subdivided into sufficiently small segments because the number of segments would exceed that allowed by the dimensioned variables.  Presently, 63 segments are allowed per section.)

If LNGSAM=T, this variable is read only for the first crack (M=1).  Note that the maximum number of subdivided sections for all the cracks is 108.  An example with 3 sections is given in Figure 2.

4. Read:  WPU:

This statement is executed for the first crack if all cracks have the same width (LWSAME=T).  If LWSAME=F, this statement is executed for each crack.

a) WPU:  This real variable specifies the width of the M$^{th}$ crack. The maximum width of the crack should not exceed 0.1 wavelength.
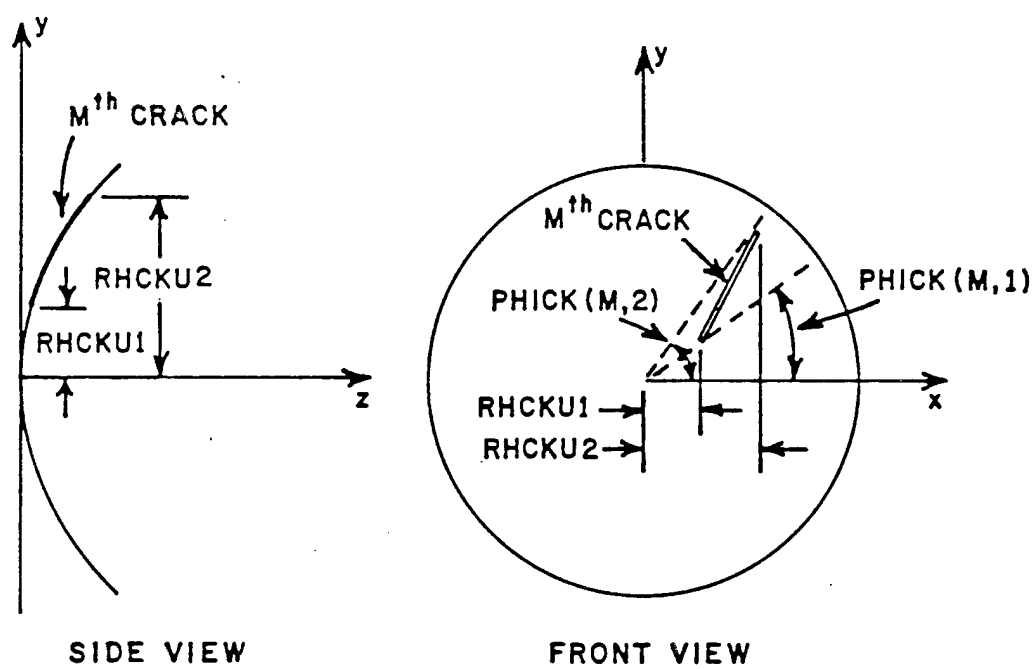
Figure 1. Coordinate system for the end points of the M<sup>th</sup> crack.
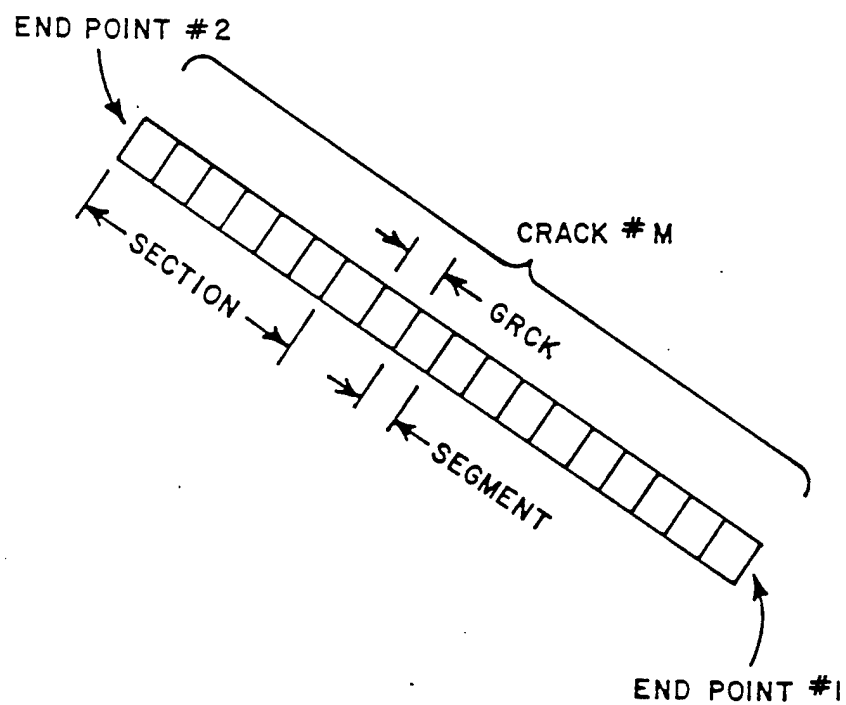
Figure 2. Example of a crack divided into 3 sections NCC(M)=3. In this particular example each section is divided into 6 segments by the choice of GRCK.

```
------------------------------------------------------------
Command BS:  DATA FOR NEC-BSC
------------------------------------------------------------
```

This command enables the user to use the NEC Basic Scattering Code [11] to handle more complicated scatterers, such as cylinders, in the near field of the reflector antenna. Two models have been developed for use with the Basic Scattering Code, namely, the YSUM model and the 2-dimensional array model. If either model is used, the reflector code will generate SG:, XQ:, and EN: commands for the Basic Scattering Code input and store it in the write Unit #2. The user should include these commands in his input file for the Basic Scattering Code.

1. Read: LBS1,LBS2,HS

   a) LBS1: If this logical variable is set true the YSUM model is used. Since the YSUM model treats a Y-integration equivalent line source as a single dipole source, this model is quite efficient but restricted to the cases where the plate or cylinder scatterers are long with their axes vertically oriented and the far field pattern is in the PHI= 0 plane. The upper and lower edges of the scatterer should lie outside the projected aperture as shown in Figure 1 of the TP: Command.

   b) LBS2: If this logical variable is set true (and LBS1 is set false) the 2D array model is used. With this model the reflector is represented as a 2D dipole array source which uses the aperture field over the reflector grid. Note that although this model is more accurate than the YSUM model, it is very time consuming because a large number of sources must often be used. Currently the maximum number of sources is limited to 30 by the array dimension of the Basic Scattering Code.

      NOTE: LBS1 and LBS2 should not be true at the same time. If that happens, the YSUM model will be used.

   c) HS: This real variable is used to input the length of all the dipole elements. The unit of HS is specified by the US: Command of the Basic Scattering Cqde. Usually, HS=0.2 wavelength is a good choice.

NOTE: The BS: Command capability is currently restricted to y-polarized electric fields in the aperture, i.e., TAU= 90° in the FD: Command.

# REFERENCES

[1]  R.C. Rudduck and Y.C. Chang, "Numerical Electromagnetic Code -
     Reflector Antenna Code, NEC-REF (Version 2), Part I:  User's
     Manual," Report 712242-16, December 1982, The Ohio State
     University ElectroScience Laboratory, Department of Electrical
     Engineering; prepared under Contract No. N00123-79-C-1469 for
     Naval Regional Procurement Office.

[2]  Y.C. Chang and R.C. Rudduck, "Numerical Electromagnetic Code -
     Reflector Antenna Code, NEC-REF (Version 2), Part II:  Code
     Manual," Report 712242-17, December 1982, The Ohio State
     University ElectroScience Laboratory, Department of Electrical
     Engineering; prepared under Contract No. N00123-79-C-1469 for
     Naval Regional Procurement Office.

[3]  S.H. Lee, "GTD Analysis of Reflector Antennas with General Rim
     Shapes - Near and Far Field Solutions," Ph.D. Dissertation and
     Report 712242-4 (713742), The Ohio State University ElectroScience
     Laboratory, Department of Electrical Engineering; prepared under
     Contract No. N00123-79-C-1469 for Naval Regional Procurement
     Office, September 1980.

[4]  R.J. Marhefka and W.D. Burnside, "Numerical Electromagnetic Code
     (NEC) - Basic Scattering Code, Part I:  User's Manual (Version
     2)," Report 712242-14 (713742), The Ohio State University
     ElectroScience Laboratory, Department of Electrical Engineering;
     prepared under Contract No. N00123-79-C-1469 for Naval Regional
     Procurement Office, September 1980.

[5]  R.G. Kouyoumjian and P.H. Pathak, "A Uniform Geometrical Theory of
     Diffraction for an Edge of a Perfectly Conducting Surface," Proc.
     Inst. Elec. Eng., Vol. 62, No. 11, pp. 1448-1461, November 1974.

[6]  W.D. Burnside, Nan Wang and E.L. Pelton, "Near Field Pattern
     Analysis of Airborne Antennas," IEEE Trans. Antennas and
     Propagation, Vol. AP-28, May 1980.

[7]  Y.C. Chang, "Analysis of Reflector Antennas with Array Feeds Using
     Multi-Point GTD and Extended Aperture Integration," Technical
     Report 715559-3, prepared under Contract NAS1-17450, for
     NASA/Langley Research Center, Hampton, Virginia, March 1984.

[8]  S.H. Lee, R.C. Rudduck, C.A. Klein, and R.G. Kouyoumjian, "A GTD
     Analysis of the Circular Reflector Including Feed and Strut
     Scatter," Technical Report 4381-1, The Ohio State University
     ElectroScience Laboratory, May 1977.

[9]   T.H. Lee and R.C. Rudduck, "Geometrical Optics and GTD Analysis of Subreflector in Cassegrain and Gregorian Reflector Antennas," 713712-2

[10]  A.C. Ludwig, "The Definition of Cross Polarization," IEEE Trans. on Antennas and Propagation, AP-21, pp. 116-119, January 1973.

[11]  R.J. Marhefka and W.D. Burnside, "Numerical Electromagnetic Code (NEC) - Basic Scattering Code, Part I:  User's Manual (Version 2)," Report 712242-14 (713742), The Ohio State University ElectroScience Laboratory, Department of Electrical Engineering; prepared under Contract No. N00123-79-C-1469 for Naval Regional Procurement Office, October 1982.

[12]  P.A.J. Ratnasiri, R.G. Kouyoumjian and P.H. Pathak, "The Wide Angle Side Lobes of Reflector Antennas," Report 2183-1, The Ohio State University ElectroScience Laboratory, Department of Electrical Engineering; prepared under Contract AF 19(628)-5929 for Air Force Cambridge Research Laboratory, March 1970.

[13]  H.P. Colement, R.M. Brown, and G.D. Wright, "Paraboloidal Reflector Offset Fed with a Corrugated Conical Horn," IEEE Transactions on Antennas and Propagation, Vol. 23, pp. 817-819, November 1975.

[14]  D.C. Hogg, F.O. Guiraud, J. Howard, A.C. Newell, D.P. Kremer and A.G. Repjar, "An Antenna for Dual-Wavelength Radiometry at 21 and 32 GHz," IEEE Trans. Antennas and Propagation, Vol. AP-27, pp. 764-771, November 1979.

# APPENDIX A

## ANALYTIC FUNCTIONS FOR FEED PATTERNS

The reflector antenna code uses either a piecewise linear feed pattern, or the analytic functions described below. For sum patterns, the analytic function is given by

$$f_\Sigma(\psi) = \frac{e^{-A\left(\frac{\psi}{\psi_0}\right)^2}\left[\cos^N\left(\frac{\pi\psi}{2\psi_0}\right)\right] + C}{1 + C} \quad \text{ISYM} > 0 \qquad (A.1)$$

where the constants A, $\psi_0$ and C can be controlled for each input pattern cut $\psi_n$. The pattern value in Equation (A.1) is normalized such that $f_\Sigma(0)=1$ for all $\phi$-plane cuts. The constants A, C and N control the shape of the pattern. The constant $\psi_0$ permits a given pattern shape to be stretched or compressed.

For large values of $\frac{\psi}{\psi_0}$, $f_\Sigma(\psi) \rightarrow \frac{C}{1+C}$. In many cases, this represents a spillover level that is too high for typical feed patterns. Consequently, the feed subroutine uses a linear taper under certain conditions for $\psi_{L1} < \psi < \psi_{L2}$ as shown in Figure 1, where $\psi_{L1}$ and $\psi_{L2}$ are input in the FD: Command and are controlled by logical variable LPSIL. If LPSIL=F, Equation (A.1) is used for the entire feed pattern cut.

For difference feed patterns, the analytic function is given by

$$f_\Delta(\psi) = C\, e^{-A\left(\frac{\psi}{\psi_0}\right)^2} \sin^N\left(\frac{\pi\psi}{2\psi_0}\right) \qquad \text{ISYM} < 0 \qquad (A.2)$$

for the entire feed pattern cut.

The parameters in Equations (A.1) and (A.2) correspond to the following input variables used in the code. Refer to the FD: Command.

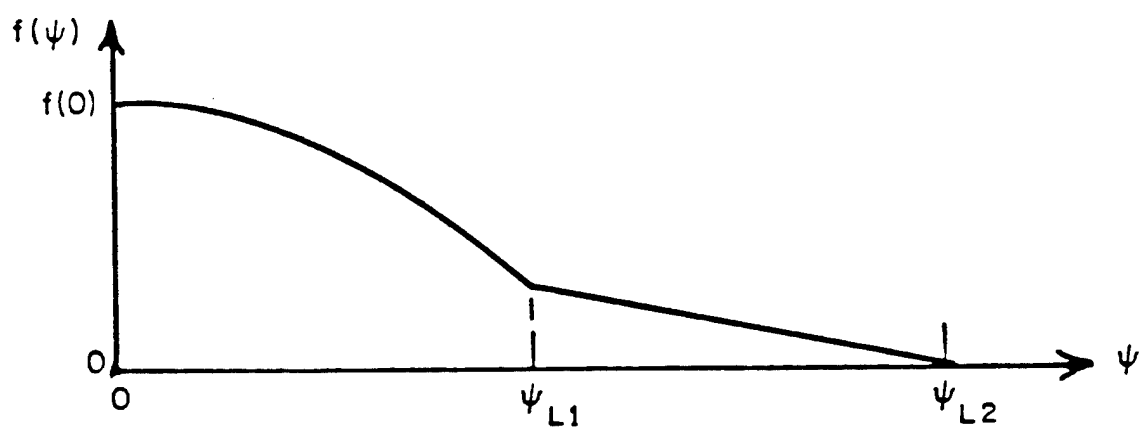| Parameter | Code Name |
|-----------|-----------|
| N | NPW |
| A | AEX |
| C | CAN |
| $\psi_0$ | PSIO |

A-1

Figure 1. Analytic feed pattern with linear taper region.

## APPENDIX B

## EXAMPLES OF A 24" FOCAL POINT REFLECTOR

In this section, several examples of a 24" focal point parabolic
reflector antenna are given to show how the basic commands are used to
calculate the patterns of the antenna. The geometry of this reflector
with an on-axis feed is shown in Figure 1. Calculated results are given
for a frequency of 11.0 GHz. A GTD analysis was previously developed
for the far field pattern of this circular reflector as reported in
Reference [12].

EXAMPLE 1

This example shows the default case of the Reflector Antenna Code.
A 24" circular reflector antenna with 8" focal distance is chosen as the
default reflector, and a focused feed with an analytic feed pattern
(LLFD=F) is chosen as the default feed. Feed blockage, strut and plate
are not included in the default case. The only input needed to run this
case is the XQ: Command. The line printer output is shown in Table 1.
Note that a 5° increment is used in the line printer output. Normally,
about a 0.5° increment would be used for plotting purposes. The plot of
the H-plane pattern is shown in Figure 2. To obtain this plot, one has
to input PZ: and PP: Commands before the XQ: Command.
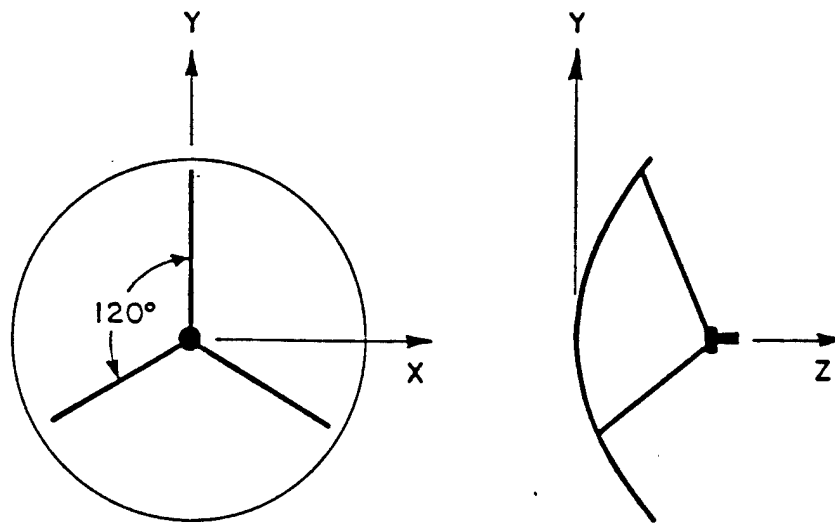
(a) FRONT VIEW          (b) SIDE VIEW

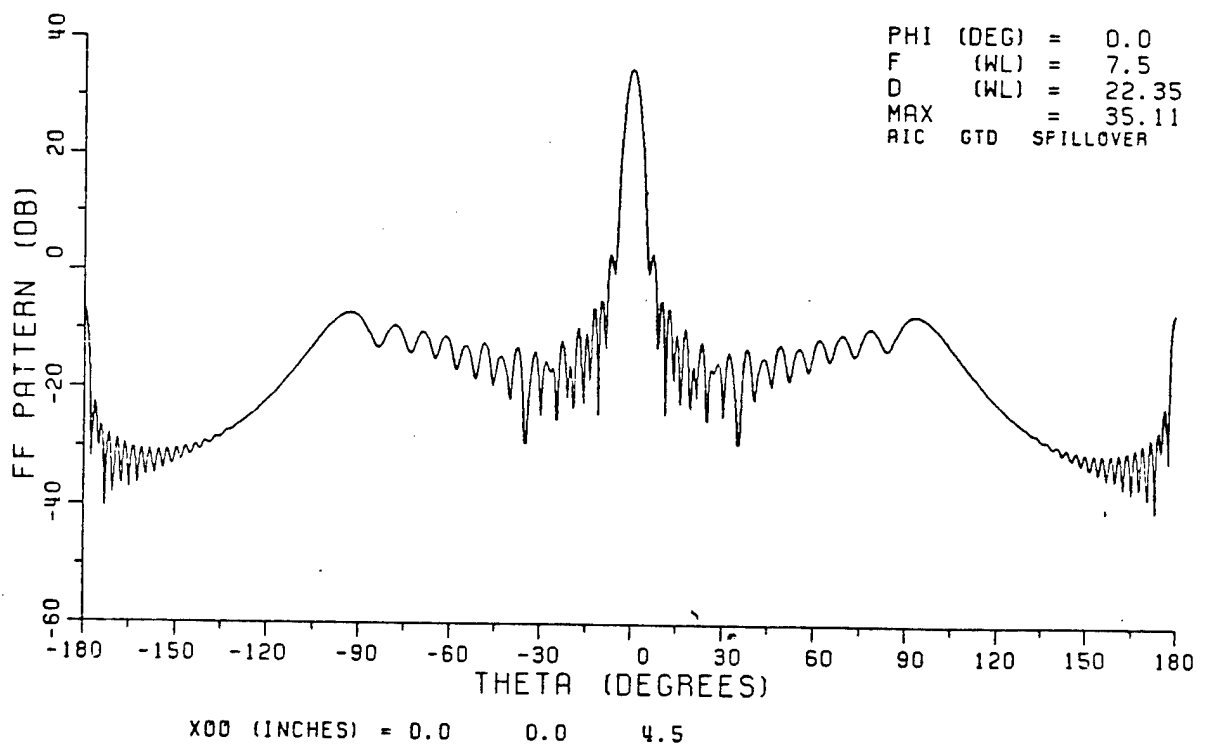Figure 1. Circular reflector antenna with an on-axis feed.



PHI  (DEG)  =   0.0
F    (WL)   =   7.5
D    (WL)   =   22.35
MAX         =   35.11
AIC   GTD   SPILLOVER

XDD (INCHES) = 0.0      0.0      4.5

Figure 2. H-plane pattern of the circular reflector antenna calculated
by OSU Reflector Antenna Code with default analytic feed.

## TABLE 1

## LINE PRINTER OUTPUT OF EXAMPLE 1

****** OUTPUT LISTING FROM THE OSU REFLECTOR ANTENNA CODE ( NASL4 ) ******

```
*********************************************************************************
*                                                                              *
*                    DEFAULT DATA                                              *
*                                                                              *
*                                                                              *
* NTYPE= 1                                                                     *
* LINEAR DIMENSION INPUTS ARE IN INCHES                                        *
*                                                                              *
*     CIRCULAR REFLECTOR WITH APERTURE DIAMETER =    24.00                     *
*                                                                              *
*                                                                              *
*     FOCAL DISTANCE=      8.00 GRIDX= 0.600       GRIDY = 0.600               *
*                                                                              *
*     FEED PATTERN SYMMETRY GIVEN BY:ISYM= 1                                   *
*                                                                              *
*                                                                              *
*     LINEARLY POLARIZED FEED                                                  *
*                                                                              *
*       POLARIZED ANGLE = 90.00                                               *
*                                                                              *
*       ANALYTIC FEED DATA                                                     *
*                                                                              *
*                                                                              *
*         NPW = 1                                                              *
*           N          PHIN(N)      PSIO(N)        AEX(N)       CAN(N)         *
*           1           0.0         120.0           5.0         0.09          *
*           2          90.0         140.0           6.0         0.10          *
*                                                                              *
*                                                                              *
*********************************************************************************


*********************************************************************************
*                                                                              *
*   XQ:                                                                        *
*                                                                              *
*                                                                              *
*                                                                              *
*                                                                              *
*                                                                              *
*                                                                              *
*                                                                              *
*                                                                              *
*     FREQUENCY =    11.000 GHZ                                                *
*                                                                              *
*     WAVELENGTH =    0.027273 METERS                                          *
*     * THE FOLLOWING DIMENSION UNITS ARE IN WAVELENGTHS *                     *
*     * ANGLES & PHASE ARE IN DEGREES *                                        *
*                                                                              *
*                                                                              *
*     APERTURE DIAMETER =    22.35 WAVELENGTHS                                 *
*                                                                              *
*     NUMBER OF RIM SEGMENTS= 96                                               *
*                                                                              *
*                                                                              *
*     FOCAL DISTANCE =      7.45                                               *
*                                                                              *
*     GRIDX =   0.56      GRIDY =   0.56                                       *
RGEOM:RHOS1,BOUND=   11.73480        13.42983
```

TABLE 1 - CONTINUED

```
ZOP=    4.191000
      DISTANCE FROM FOCUS TO RIM: RO=    11.642

      FEED LOCATED AT (       0.00,      0.00,      7.45,)
                   ( REFERRED TO VERTEX )


      FEED POWER: PRAD = 0.138E+01

      FAR FIELD GAIN REF =     -7.859

      NUMBER OF PRINCIPAL GRID LINES: MMAX= 43      NMAX= 43

      APERTURE CENTER AT (   0.000,   0.000,          4.191 )




      CPU TIME FOR APERTURE FIELD CALCULATION =  10.02 SECONDS


      SHADOW BOUNDARY ANGLES: TH1 =  253.73     TH2 =  106.27
      THEB (DEG) =   90.000
      NUMBER OF ROTATED GRID LINES: IMAX= 41      JMAX= 41



      NP=  1   P2=     0.00
      CPU TIME FOR Y-INTEGRATION =    3.19 SECONDS


      AI/GTD SWITCHOVER PARAMETERS:
      THETAX =   12.21      ZX =          0.000
         NT   =   73      P3X =    10.000
         NGTD1=   34      PG1I= -180.000
         NAI  =    5      PAI =  -10.000
         NGTD2=   34      PG2I=   15.000

 PHI =    0.00
```

| THETA | PRINCIPAL POL | | | CROSS POL | | |
|---|---|---|---|---|---|---|
| | MAG | DB | PHASE | MAG | DB | PHASE |
| -180.00 | 0.113E+01 | -6.77 | 130.2 | 0.108E-05 | -127.16 | 154.4 |
| -175.00 | 0.746E-01 | -30.40 | 121.9 | 0.278E-05 | -118.97 | -5.5 |
| -170.00 | 0.438E-01 | -35.03 | 100.3 | 0.873E-06 | -129.04 | 6.5 |
| -165.00 | 0.338E-01 | -37.27 | 58.4 | 0.416E-06 | -135.49 | 30.0 |
| -160.00 | 0.443E-01 | -34.94 | 11.5 | 0.275E-06 | -139.08 | 69.8 |
| -155.00 | 0.643E-01 | -31.70 | -21.1 | 0.241E-06 | -140.22 | 108.4 |
| -150.00 | 0.724E-01 | -30.67 | -55.6 | 0.223E-06 | -140.91 | 138.3 |
| -145.00 | 0.653E-01 | -31.56 | -119.6 | 0.881E-07 | -148.96 | -172.3 |
| -140.00 | 0.845E-01 | -29.33 | 155.2 | 0.136E-06 | -145.19 | -59.2 |
| -135.00 | 0.928E-01 | -28.51 | 57.7 | 0.603E-07 | -152.25 | 13.8 |
| -130.00 | 0.116E+00 | -26.56 | -63.5 | 0.109E-06 | -147.09 | 144.0 |
| -125.00 | 0.145E+00 | -24.60 | 152.9 | 0.343E-07 | -157.14 | -20.0 |
| -120.00 | 0.192E+00 | -22.20 | -13.5 | 0.306E-07 | -158.14 | 177.4 |
| -115.00 | 0.266E+00 | -19.37 | 157.0 | 0.207E-07 | -161.55 | 0.6 |
| -110.00 | 0.384E+00 | -16.17 | -54.7 | 0.146E-07 | -164.54 | -110.3 |
| -105.00 | 0.565E+00 | -12.81 | 73.0 | 0.353E-07 | -156.90 | 62.2 |
| -100.00 | 0.819E+00 | -9.59 | -179.2 | 0.651E-07 | -151.59 | -137.3 |
| -95.00 | 0.104E+01 | -7.55 | -87.7 | 0.128E-06 | -145.73 | -66.1 |

TABLE 1 - CONTINUED

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| W | -90.00 | 0.972E+00 | -8.10 | -6.1 | 0.117E-06 | -146.49 | -0.3 | W |
| W | -85.00 | 0.629E+00 | -11.89 | 93.9 | 0.338E-07 | -157.27 | 43.4 | W |
| W | -80.00 | 0.872E+00 | -9.05 | -147.0 | 0.154E-06 | -144.11 | -131.7 | W |
| W | -75.00 | 0.740E+00 | -10.47 | -68.1 | 0.132E-06 | -145.44 | -126.9 | W |
| W | -70.00 | 0.894E+00 | -8.83 | 46.6 | 0.173E-06 | -143.08 | 55.4 | W |
| W | -65.00 | 0.662E+00 | -11.44 | 137.0 | 0.167E-07 | -163.43 | -72.3 | W |
| W | -60.00 | 0.817E+00 | -9.62 | -140.6 | 0.128E-06 | -145.68 | -171.0 | W |
| W | -55.00 | 0.873E+00 | -9.04 | -45.5 | 0.154E-06 | -144.13 | -38.1 | W |
| W | -50.00 | 0.846E+00 | -9.32 | 41.0 | 0.166E-06 | -143.46 | 71.8 | W |
| W | -45.00 | 0.710E+00 | -10.84 | 124.6 | 0.208E-06 | -141.52 | -149.7 | W |
| W | -40.00 | 0.657E+00 | -11.51 | -179.3 | 0.101E-06 | -147.81 | -28.6 | W |
| W | -35.00 | 0.524E+00 | -13.46 | -110.9 | 0.237E-06 | -140.35 | 43.9 | W |
| W | -30.00 | 0.654E+00 | -11.55 | -58.2 | 0.166E-06 | -143.47 | 99.8 | W |
| W | -25.00 | 0.594E+00 | -12.38 | -29.3 | 0.269E-06 | -139.27 | -176.2 | W |
| W | -20.00 | 0.299E+00 | -18.36 | -1.6 | 0.500E-06 | -133.88 | -157.1 | W |
| W | -15.00 | 0.178E+00 | -22.84 | 162.3 | 0.853E-06 | -129.24 | -138.8 | W |
| W | -10.00 | 0.974E+00 | -8.08 | -171.4 | 0.111E-05 | -126.99 | 139.0 | W |
| W | -5.00 | 0.398E+01 | 4.14 | 47.7 | 0.153E-05 | -124.19 | 107.3 | W |
| W | 0.00 | 0.138E+03 | 34.96 | 39.0 | 0.112E-04 | -106.87 | 39.0 | W |
| W | 5.00 | 0.398E+01 | 4.14 | 47.7 | 0.144E-05 | -124.69 | -28.1 | W |
| W | 10.00 | 0.974E+00 | -8.08 | -171.4 | 0.103E-05 | -127.60 | -61.7 | W |
| W | 15.00 | 0.178E+00 | -22.84 | 162.3 | 0.280E-06 | -138.91 | 39.9 | W |
| W | 20.00 | 0.299E+00 | -18.36 | -1.6 | 0.141E-06 | -144.87 | 28.9 | W |
| W | 25.00 | 0.594E+00 | -12.38 | -29.3 | 0.138E-06 | -145.05 | -12.1 | W |
| W | 30.00 | 0.654E+00 | -11.55 | -58.2 | 0.994E-07 | -147.91 | -71.5 | W |
| W | 35.00 | 0.524E+00 | -13.47 | -110.9 | 0.126E-06 | -145.84 | -126.9 | W |
| W | 40.00 | 0.657E+00 | -11.51 | -179.3 | 0.104E-06 | -147.51 | 173.5 | W |
| W | 45.00 | 0.710E+00 | -10.84 | 124.6 | 0.122E-06 | -146.15 | 68.8 | W |
| W | 50.00 | 0.846E+00 | -9.32 | 41.0 | 0.567E-07 | -152.78 | -18.5 | W |
| W | 55.00 | 0.873E+00 | -9.04 | -45.5 | 0.378E-07 | -156.30 | -68.7 | W |
| W | 60.00 | 0.817E+00 | -9.62 | -140.6 | 0.380E-07 | -156.27 | -99.1 | W |
| W | 65.00 | 0.662E+00 | -11.44 | 137.0 | 0.759E-07 | -150.26 | 134.3 | W |
| W | 70.00 | 0.894E+00 | -8.83 | 46.6 | 0.139E-07 | -165.01 | -102.7 | W |
| W | 75.00 | 0.740E+00 | -10.47 | -68.1 | 0.102E-06 | -147.70 | 0.7 | W |
| W | 80.00 | 0.872E+00 | -9.05 | -147.0 | 0.243E-07 | -160.14 | 106.1 | W |
| W | 85.00 | 0.629E+00 | -11.89 | 93.9 | 0.648E-07 | -151.63 | 107.7 | W |
| W | 90.00 | 0.972E+00 | -8.10 | -6.1 | 0.474E-07 | -154.34 | -23.1 | W |
| W | 95.00 | 0.104E+01 | -7.55 | -87.7 | 0.513E-07 | -153.66 | -133.8 | W |
| W | 100.00 | 0.819E+00 | -9.59 | -179.2 | 0.817E-07 | -149.61 | 138.3 | W |
| W | 105.00 | 0.565E+00 | -12.81 | 73.0 | 0.217E-07 | -161.14 | 31.2 | W |
| W | 110.00 | 0.384E+00 | -16.17 | -54.7 | 0.121E-06 | -146.23 | -70.6 | W |
| W | 115.00 | 0.266E+00 | -19.37 | 157.0 | 0.149E-06 | -144.41 | 146.6 | W |
| W | 120.00 | 0.192E+00 | -22.20 | -13.5 | 0.140E-06 | -144.95 | -21.6 | W |
| W | 125.00 | 0.145E+00 | -24.60 | 152.9 | 0.162E-06 | -143.68 | 146.8 | W |
| W | 130.00 | 0.116E+00 | -26.56 | -63.5 | 0.186E-06 | -142.47 | -62.4 | W |
| W | 135.00 | 0.928E-01 | -28.51 | 57.7 | 0.128E-06 | -145.74 | 60.3 | W |
| W | 140.00 | 0.845E-01 | -29.33 | 155.2 | 0.239E-06 | -140.30 | 145.7 | W |
| W | 145.00 | 0.653E-01 | -31.56 | -119.6 | 0.187E-06 | -142.43 | -119.4 | W |
| W | 150.00 | 0.724E-01 | -30.67 | -55.6 | 0.322E-06 | -137.70 | -57.2 | W |
| W | 155.00 | 0.643E-01 | -31.70 | -21.1 | 0.348E-06 | -137.03 | -24.4 | W |
| W | 160.00 | 0.443E-01 | -34.94 | 11.5 | 0.390E-06 | -136.03 | 13.4 | W |
| W | 165.00 | 0.338E-01 | -37.27 | 58.4 | 0.517E-06 | -133.59 | 50.9 | W |
| W | 170.00 | 0.438E-01 | -35.03 | 100.3 | 0.101E-05 | -127.78 | 74.9 | W |
| W | 175.00 | 0.746E-01 | -30.40 | 121.9 | 0.298E-05 | -118.38 | 90.4 | W |
| W | 180.00 | 0.113E+01 | -6.77 | 130.2 | 0.174E-05 | -123.04 | 164.0 | W |

```
* -----------------------------------------------------------------
*                                                                  *
*                                                                  *
*                                                                  *
*          CPU TIME = 22.32 SECONDS                                *
*                                                                  *
*                                                                  *
****** OUTPUT LISTING FROM THE OSU REFLECTOR ANTENNA CODE (  NASL4  ) ******
*                                                                  *
*                                                                  *
*******************************************************************
```

TABLE 1 - CONTINUED

```
*************************************************************
  *************************************************************  *
 *                                                              *
 *   EN:                                                        *
 *                                                              *
 *                                                              *
 *************************************************************
```

## EXAMPLE 2

This example uses the default reflector but with a piecewise linear feed pattern (LLFD=T). The measured feed patterns for this antenna as given in Reference [12] are shown in Figure 3. The feed is linearly polarized in the y-direction. The piecewise linear feed pattern option of the reflector code was used to approximate the measured H- and E-plane feed patterns as shown in Figures 4a and 4b, respectively.

The far field patterns as computed by the reflector code are shown in Figures 5a and b for the H- and E-planes, respectively. The results from the reflector code were found to be in good agreement with the calculated results of Reference [12] without aperture blockage as shown in Figures 6a and b. Full pattern plots for $\phi=0°$, 90° and 45° are shown in Figure 7. The cross polarized component is very low in the principal planes for this case; however, the cross polarization for $\phi=45°$ is shown in Figure 7d. The input data are given in Table 2.
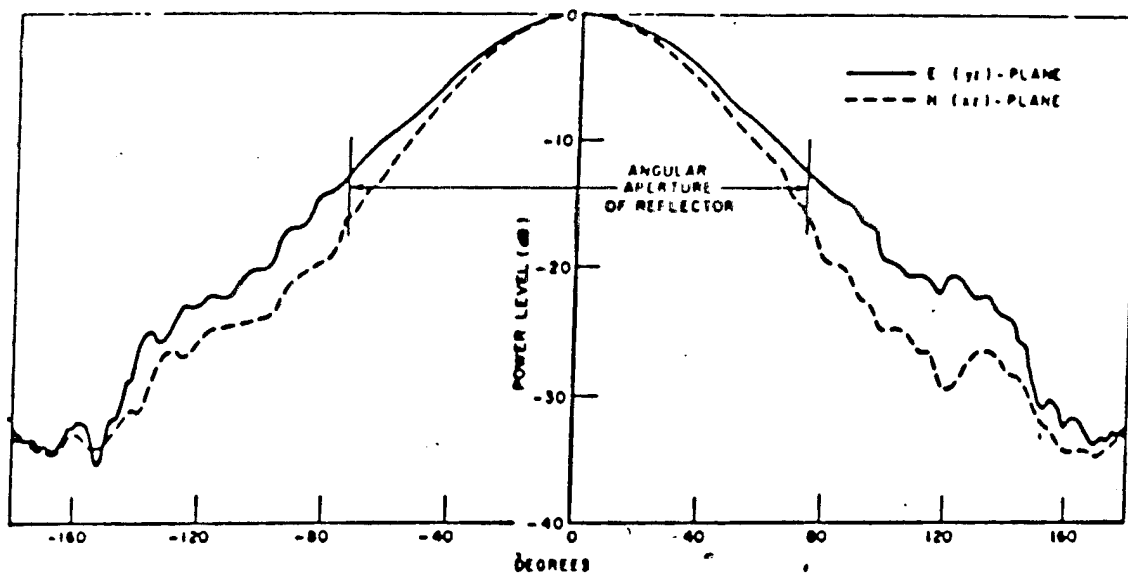


Figure 3. Measured primary field patterns of a flanged waveguide feed.

B-7
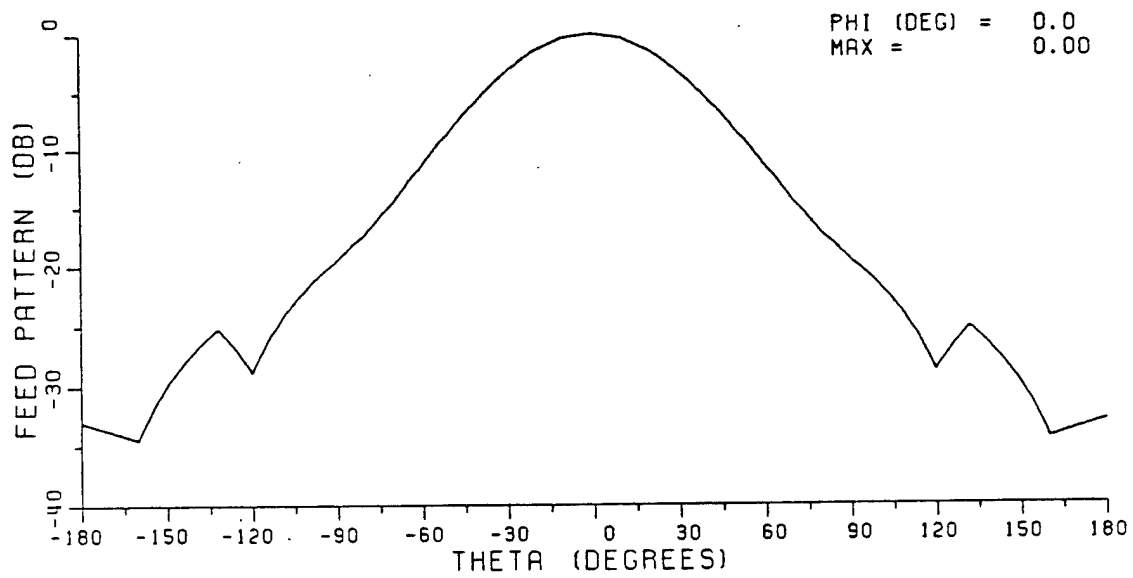
(a) H-plane
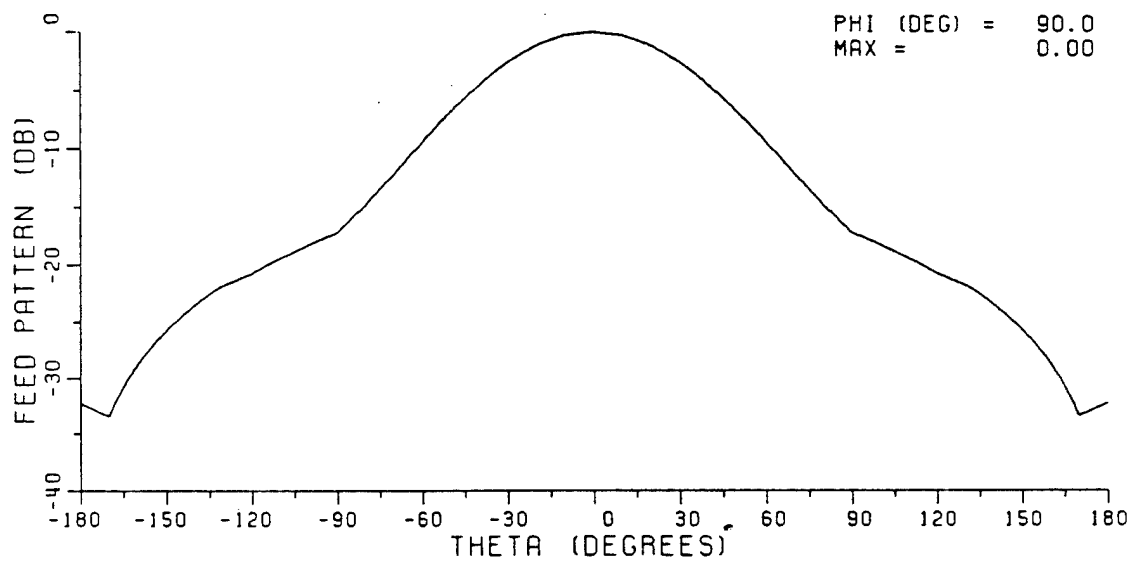


(b) E-plane

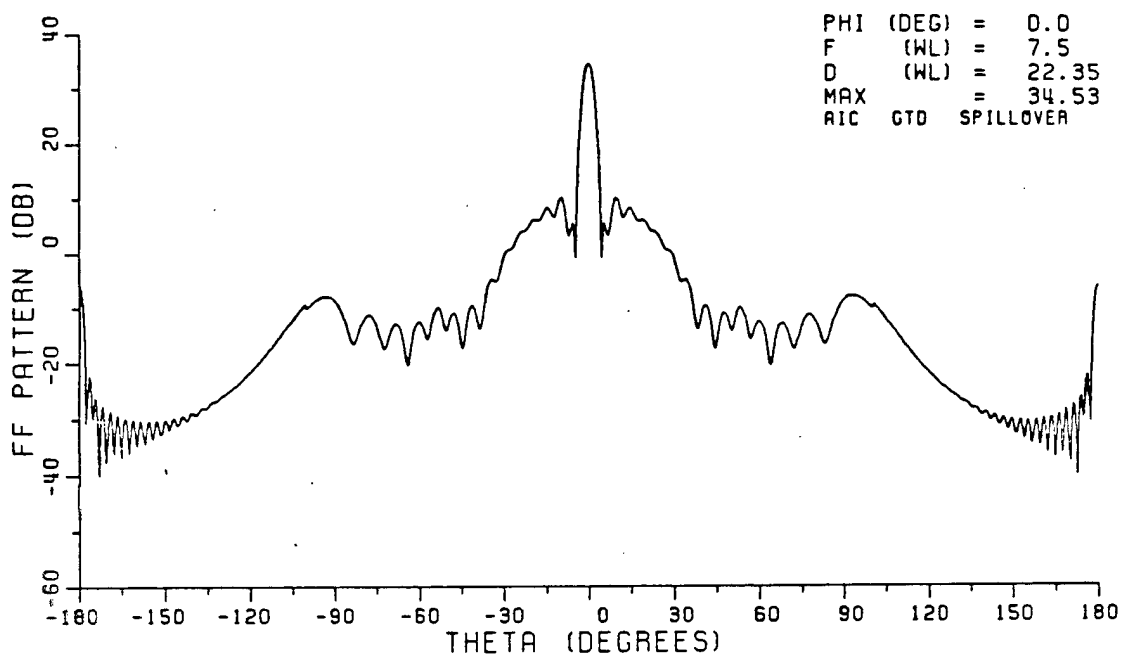Figure 4.  Input feed patterns for circular reflector Example 2.

B-8
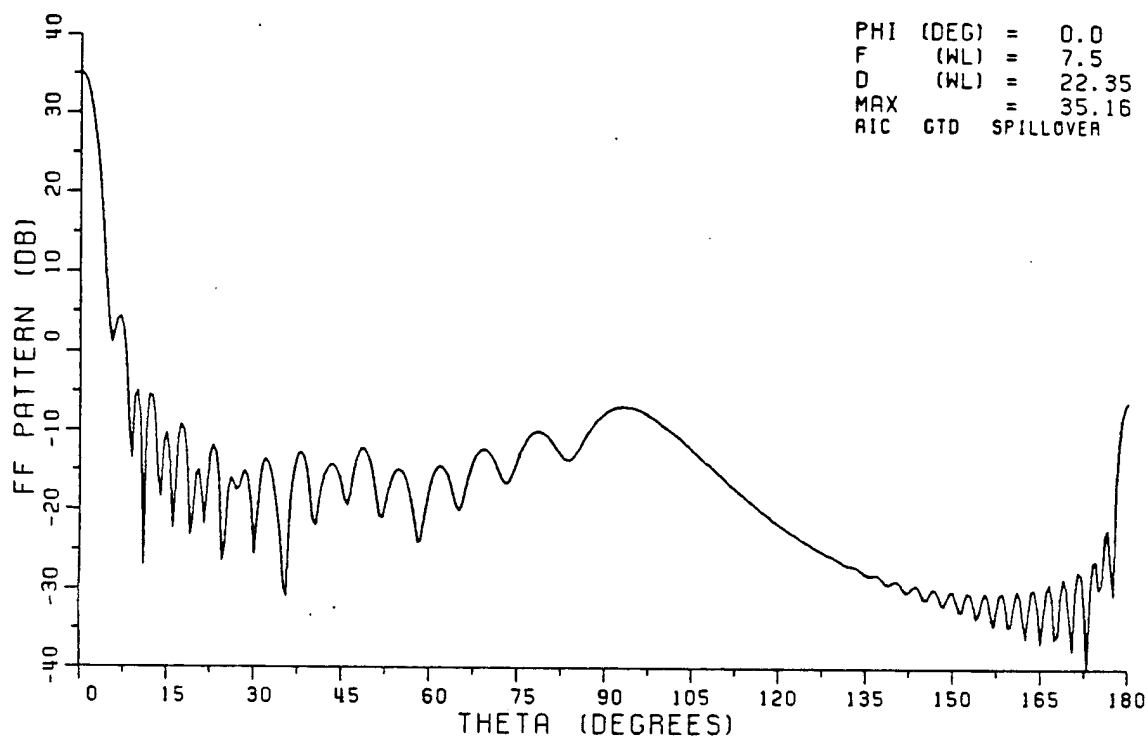
Figure 5a. Far field pattern of Example 2 computed by OSU Reflector
Antenna Code. PHI=0.0 degrees.
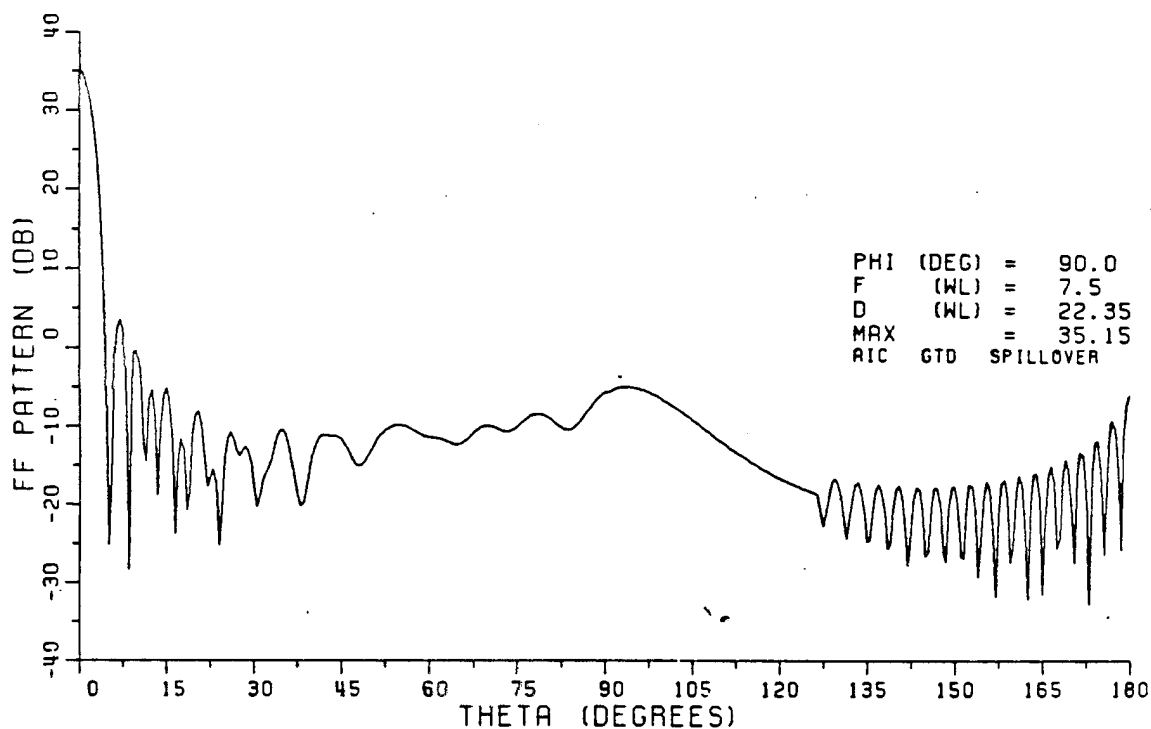


Figure 5b. Far field pattern of Example 2 computed by OSU Reflector
Antenna Code. PHI=90.0 degrees.
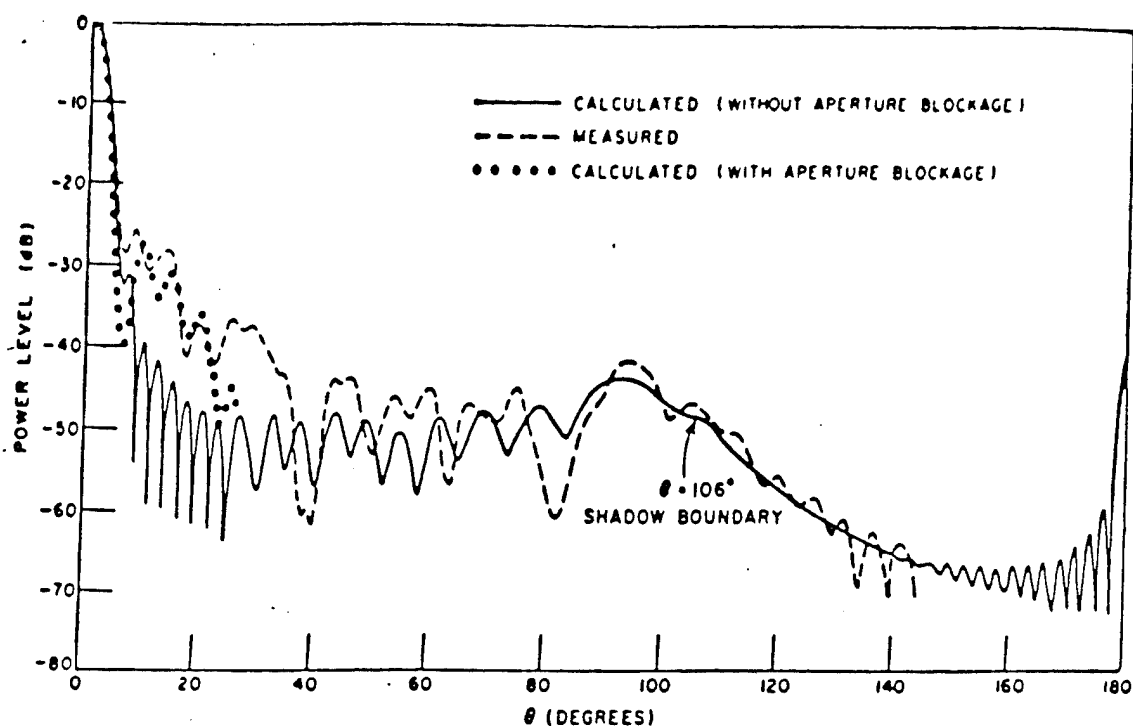
Figure 6a.   H-plane pattern of a parabolic reflector with a flanged
waveguide feed.   Computed in Reference [12].
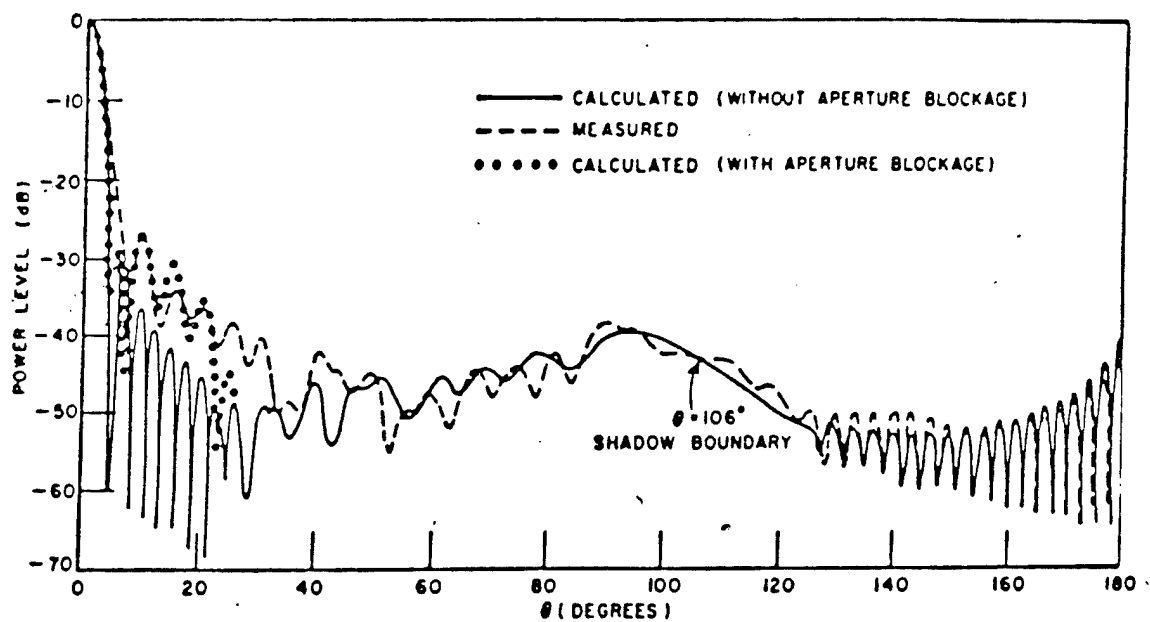


Figure 6b.   E-plane pattern of a parabolic reflector with a flanged
waveguide feed.   Computed in Reference [12].
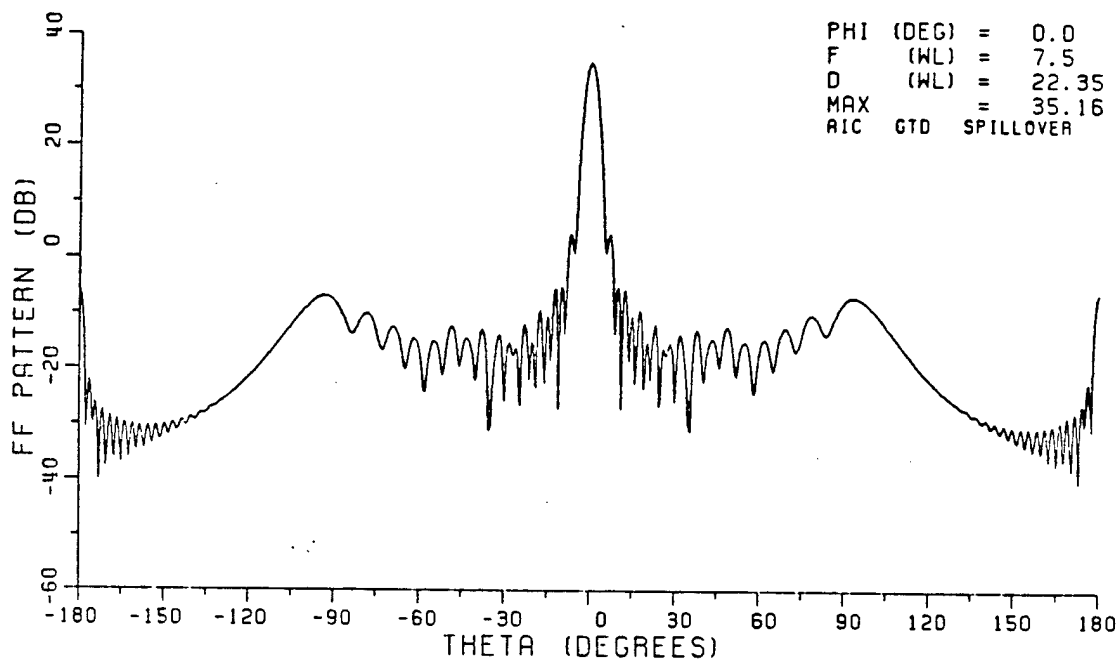
Figure 7a. Full far field pattern of Example 2 computed by OSU
Reflector Antenna Code. PHI=0.0 degrees.
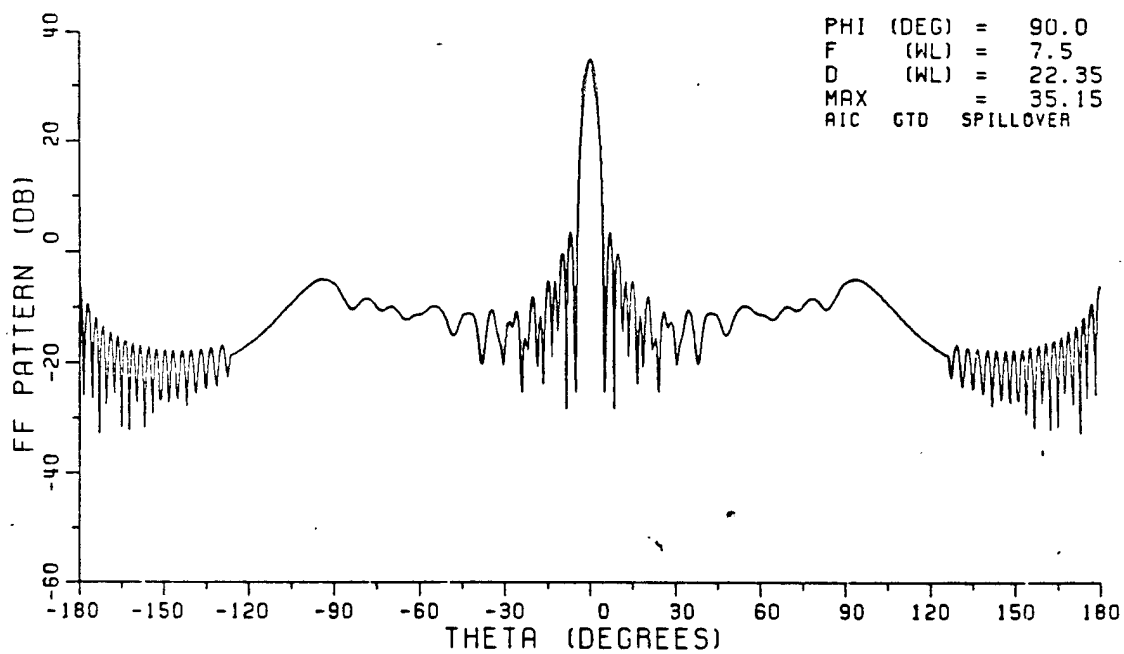


Figure 7b. Full far field pattern of Example 2 computed by OSU
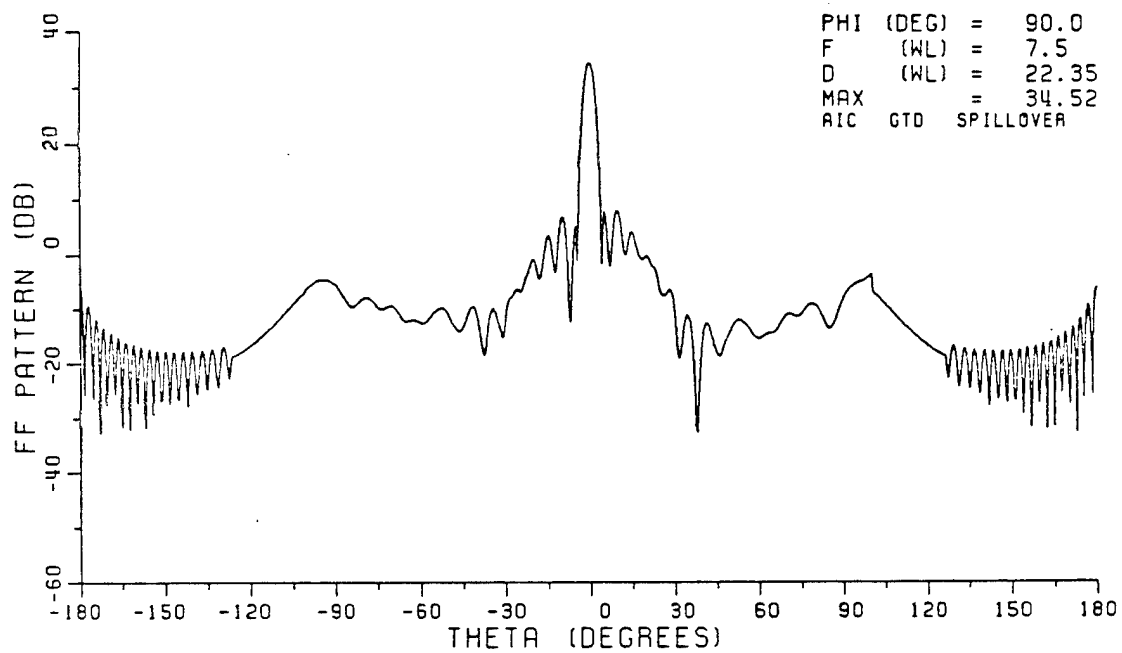Reflector Antenna Code. PHI=90.0 degrees.

B-11
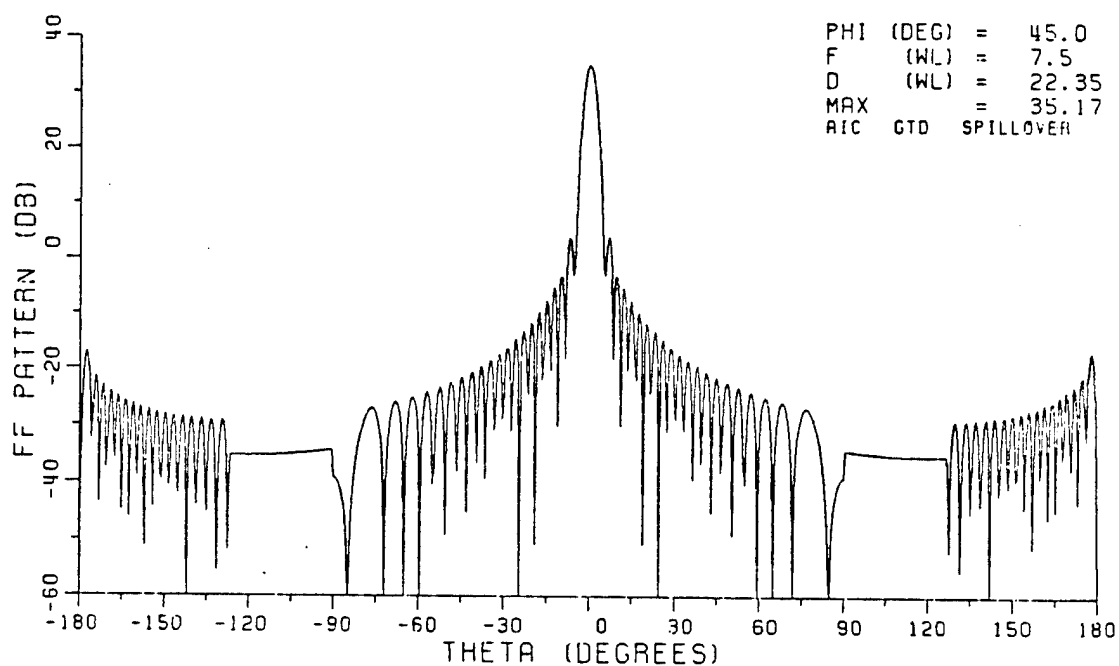
Figure 7c. Full far field pattern of Example 2 computed by OSU
Reflector Antenna Code. PHI=45.0 degrees.
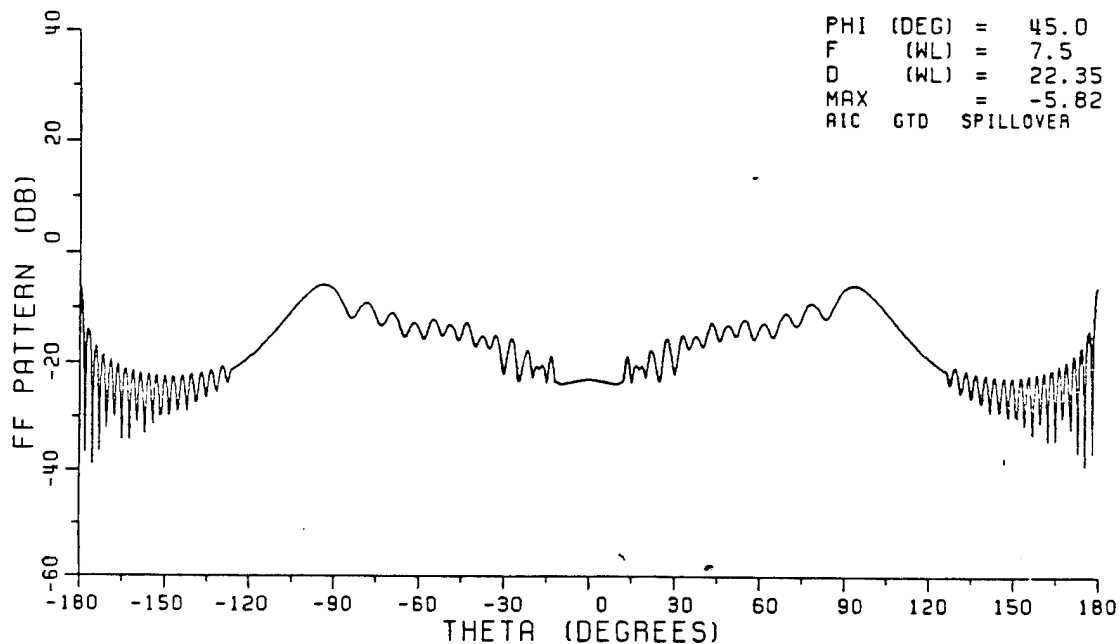


Figure 7d. Cross polarized far field pattern of Example 2 computed by
OSU Reflector Antenna Code. PHI=45.0 degrees.

B-12

# TABLE 2

## INPUT DATA FOR CIRCULAR REFLECTOR
## EXAMPLE 2

```
CM:            ***** C24B.DAT *****
CM: GENERAL EXAMPLE OF 24" CIRCULAR REFLECTOR
CE:            LINEAR FEED (LLFD-T)
DG:
1
3  8.   0.6   0.6   24.   0
FD:
0   T
T   0   F   1   90.   0.   F
2   0.   90.
14
0..        1.0000     0.    0.0000    0.
10.        0.9575     0.    0.0000    0.
20.        0.8419     0.    0.0000    0.
30.        0.6840     0.    0.0000    0.
40.        0.5200     0.    0.0000    0.
50.        0.3772     0.    0.0000    0.
60.        0.2664     0.    0.0000    0.
70.        0.1866     0.    0.0000    0.
80.        0.1358     0.    0.0000    0.
90.        0.10521    0.    0.0000    0.
120.       0.03588    0.    0.0000    0.
132.       0.05475    0.    0.0000    0.
160.       0.01884    0.    0.0000    0.
180.       0.02240    0.    0.0000    0.
0.         1.0000     0.    0.0000    0.
10.        0.9660     0.    0.0000    0.
20.        0.8714     0.    0.0000    0.
30.        0.7375     0.    0.0000    0.
40.        0.5900     0.    0.0000    0.
50.        0.4522     0.    0.0000    0.
60.        0.3360     0.    0.0000    0.
70.        0.2456     0.    0.0000    0.
80.        0.1813     0.    0.0000    0.
90.        0.13778    0.    0.0000    0.
120.       0.09170    0.    0.0000    0.
132.       0.07900    0.    0.0000    0.
170.       0.02114    0.    0.0000    0.
180.       0.02427    0.    0.0000    0.
PZ:
3
0.   45.   90.
-180. 180.   0.5
F
PP:
1
1   2
1   2
2   2
PF:
2
0.   90.
-180. 180. 1.
1
2
XQ:
```

EXAMPLE 3

This example illustrates the effects of the feed blockage and strut scattering. The antenna and feed pattern are the same as that in Example 2 except that the scattering from the feed blockage and struts as shown in Figure 1 are added in this example. Three different phi cuts at 0°, -15° and 90° are calculated and the results are shown in Figure 8. Also, the individual contributions from feed blockage and strut scattering for the $\phi=90°$ cut are shown in Figures 9a and b, respectively. Note that the strut scattering has a large effect in the $\phi=90°$ pattern cut because one of the struts is located at 90° in this case. Also note that the value of the strut boundary angle $\theta_{ST}$ is set to 100°; this causes a discontinuity at $\theta=100°$ in the 90° cut as seen in Figures 8 and 9. The input data for this case are given in Table 3.

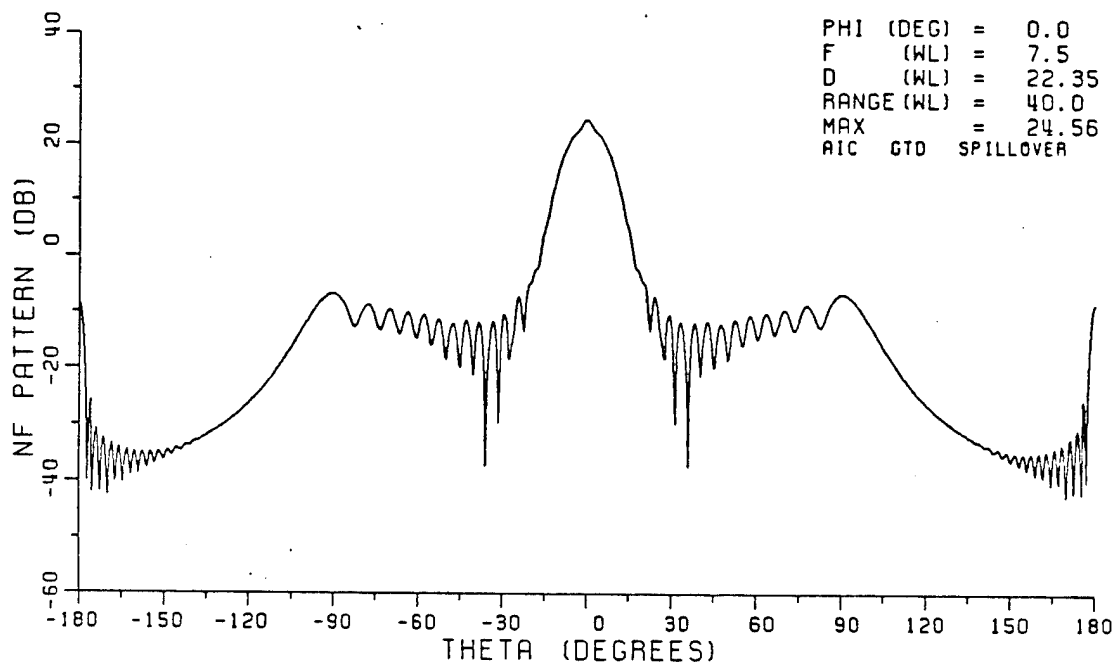Figure 8a.  Far field pattern of Example 3 computed by OSU Reflector
Antenna Code.  PHI=0.0 degrees.
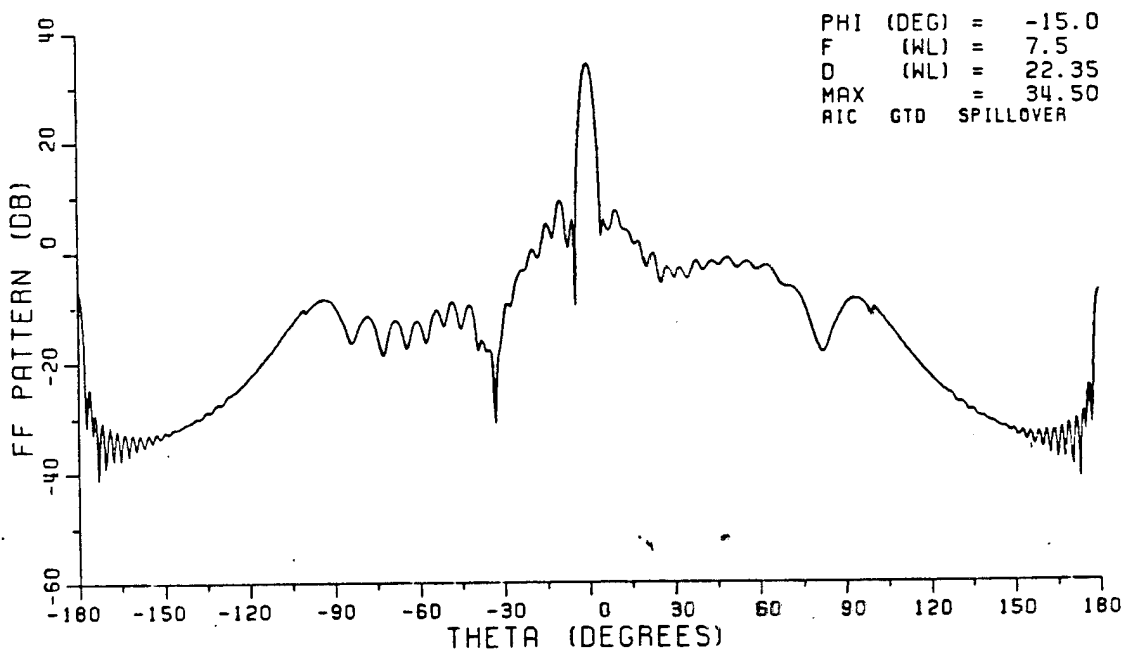


Figure 8b.  Far field pattern of Example 3 computed by OSU Reflector
Antenna Code.  PHI=-15.0 degrees.

Figure 8c. Far field pattern of Example 3 computed by OSU Reflector Antenna Code. PHI=90.0 degrees.

PHI (DEG) = 90.0
F (WL) = 7.5
D (WL) = 22.35
MAX = 4.23
AIC GTD SPILLOVER
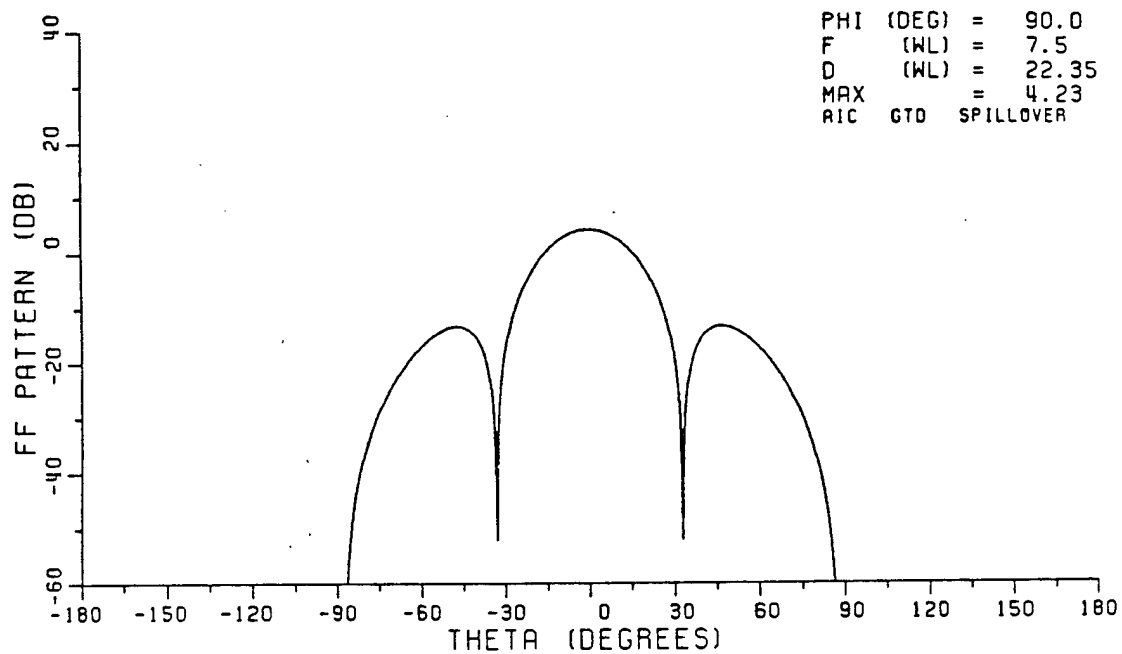
Figure 9a. Feed blockage contribution of Example 3 for PHI=90.0 degrees.



PHI (DEG) = 90.0
F (WL) = 7.5
D (WL) = 22.35
MAX = 8.07
AIC GTD SPILLOVER
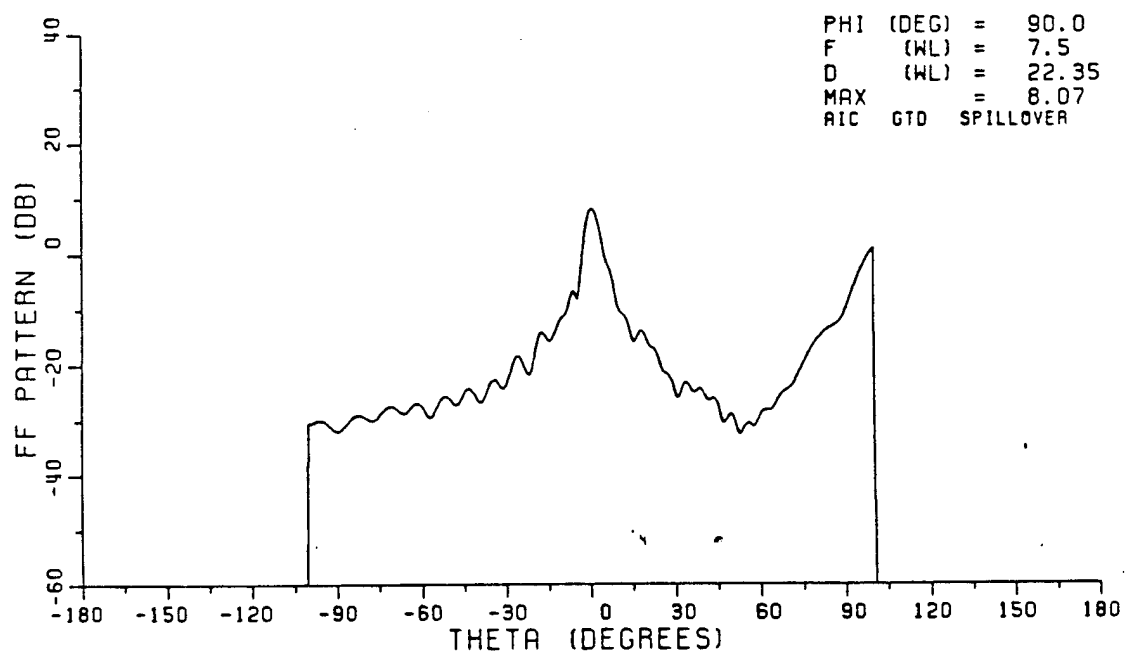
Figure 9b. Strut scattering contribution of Example 3 for PHI=90.0 degrees.

# TABLE 3

## INPUT DATA FOR THE CIRCULAR REFLECTOR
## EXAMPLE 3

```
CM:         ***** C24ST.DAT *****
CM: GENERAL EXAMPLE OF 24" CIRCULAR REFLECTOR
CM:        WITH STRUT, FEED BLOCKAGE
CE: 3 CIRCULAR STRUTS, 2.4" DIAMETER FEED BLOCKAGE
FD:
0   T
T   0   F   1   90.   0.   F
2   0.   90.
14
   0.        1.0000     0.      0.0000     0.
  10.        0.9575     0.      0.0000     0.
  20.        0.8419     0.      0.0000     0.
  30.        0.6840     0.      0.0000     0.
  40.        0.5200     0.      0.0000     0.
  50.        0.3772     0.      0.0000     0.
  60.        0.2664     0.      0.0000     0.
  70.        0.1866     0.      0.0000     0.
  80.        0.1358     0.      0.0000     0.
  90.        0.10521    0.      0.0000     0.
 120.        0.03588    0.      0.0000     0.
 132.        0.05475    0.      0.0000     0.
 160.        0.01884    0.      0.0000     0.
 180.        0.02240    0.      0.0000     0.
   0.        1.0000     0.      0.0000     0.
  10.        0.9660     0.      0.0000     0.
  20.        0.8714     0.      0.0000     0.
  30.        0.7375     0.      0.0000     0.
  40.        0.5900     0.      0.0000     0.
  50.        0.4522     0.      0.0000     0.
  60.        0.3360     0.      0.0000     0.
  70.        0.2456     0.      0.0000     0.
  80.        0.1813     0.      0.0000     0.
  90.        0.13778    0.      0.0000     0.
 120.        0.09170    0.      0.0000     0.
 132.        0.07900    0.      0.0000     0.
 170.        0.02114    0.      0.0000     0.
 180.        0.02427    0.      0.0000     0.
FB:
1   0.    0.    8.
2.4
ST:  THEST=100.                        PZ:
1                                      3
3    0.5    100.   F   T   T   T       -15. 0. 90.
10.8       90.    3.65               -180.  180.   0.5
 0.0        0.    8.00               F
3                                      PP:
0.37                                   3
10.8      210.    3.65               1   1          ! TOTAL
 0.0        0.    8.00               1   2
10.8      330.    3.65               3   1          ! FEED BLOCKAGE
 0.0        0.    8.00               1   2
                                     4   1          ! REFLECTOR/STRUT
                                     1   2
                                     XQ:
```

## EXAMPLE 4

Example 4 uses the circular reflector and analytic feed of Example 1 except that near field results are calculated instead of far field patterns. In this example, three constant range cases with R=40$\lambda$, R=100$\lambda$, and R=1000$\lambda$, are shown in Figures 10a, b, and c, respectively.

For this D=22.35$\lambda$ circular reflector, $2D^2/\lambda$=999$\lambda$, so for the R=1000$\lambda$ case, it is in the far field range. This can be verified by comparing the results in Figures 7a and 10c. The input data are given in Table 4.

PHI (DEG) =    0.0
F    (WL) =    7.5
D    (WL) =   22.35
RANGE(WL) =   40.0
MAX       =   24.56
AIC   GTD   SPILLOVER

Figure 10a.   Near field pattern of Example 4 computed by OSU Reflector
Antenna Code.   PHI=0.0 degrees, RANGE = 40.0 wavelengths.



PHI (DEG) =    0.0
F    (WL) =    7.5
D    (WL) =   22.35
RANGE(WL) =  100.0
MAX       =   30.39
AIC   GTD   SPILLOVER
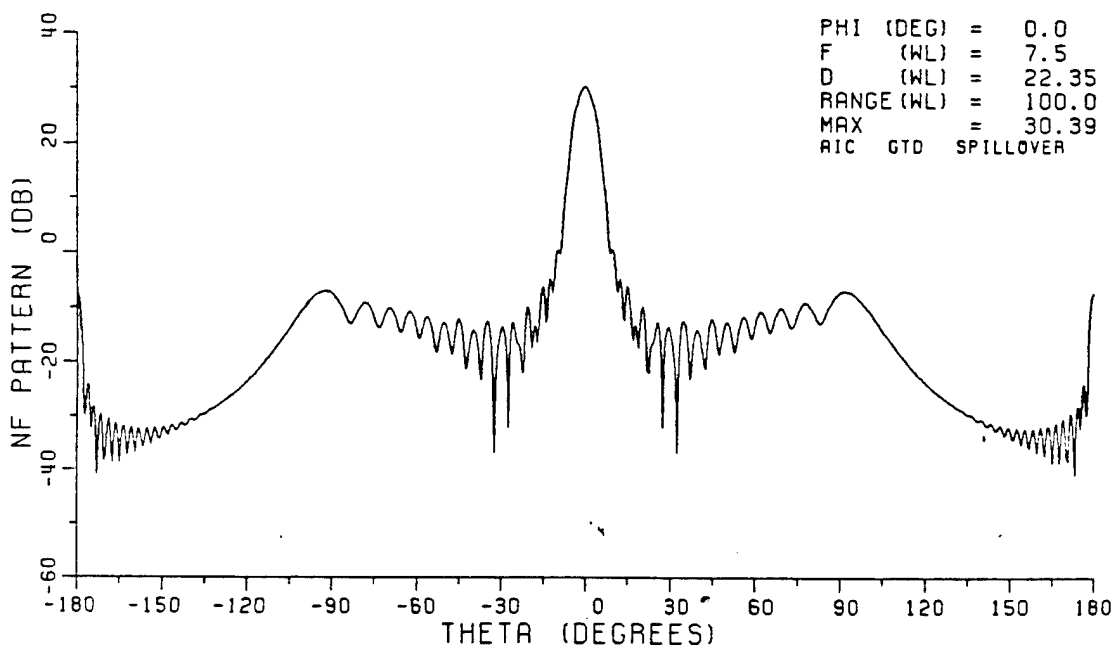
Figure 10b.   Near field pattern of Example 4 computed by OSU Reflector
Antenna Code.   PHI=0.0 degrees, RANGE = 100.0 wavelengths.

PHI (DEG) = 0.0
F (WL) = 7.5
D (WL) = 22.35
RANGE(WL) = 1000.0
MAX = 35.07
AIC GTD SPILLOVER
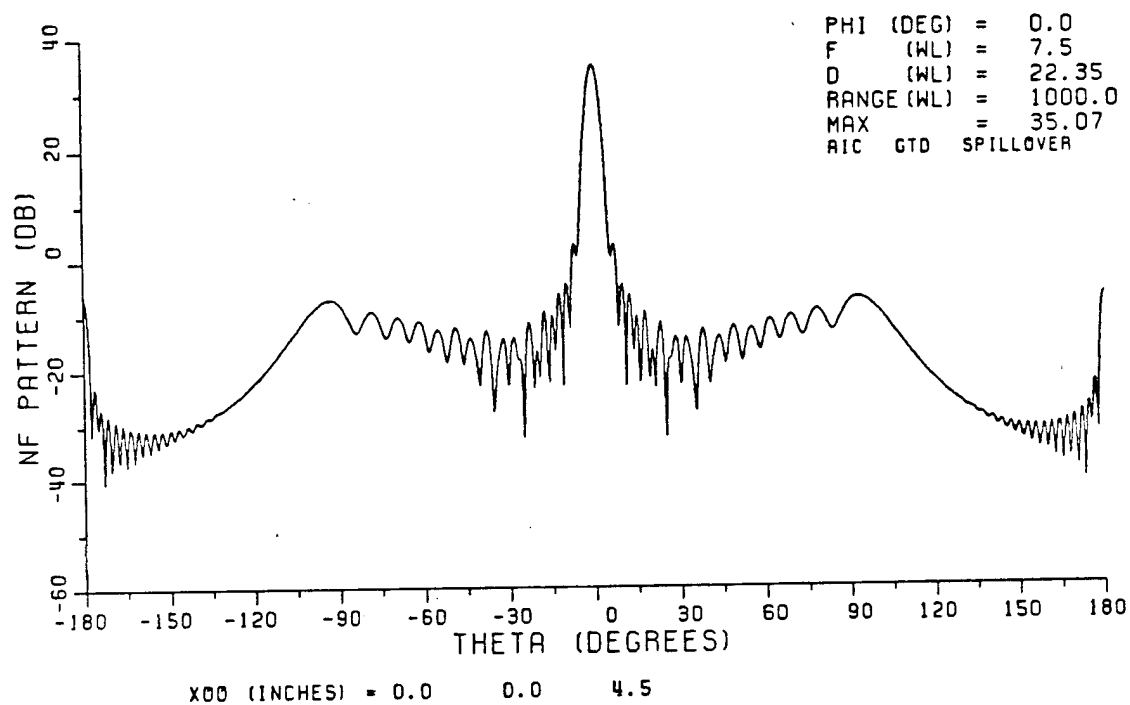
Figure 10c. Near field pattern of Example 4 computed by OSU Reflector Antenna Code. PHI=0.0 degrees, RANGE = 1000.0 wavelengths.

B-21

## TABLE 4

## INPUT DATA FOR THE CIRCULAR REFLECTOR
## EXAMPLE 4

```
CM:          ***** C24N.DAT *****
CM: GENERAL EXAMPLE OF 24" CIRCULAR REFLECTOR
CE:              NEAR FIELD
NF:
0.    0.    4.5
T
T
0.
PZ:
3
42.9497 107.3742 1073.742
-180.    180.    0.5
F
PP:
1
1  1
1  2
XQ:
```

# APPENDIX C

## EXAMPLES OF OFFSET REFLECTOR ANTENNAS

In this section, two offset reflector antennas are used as examples to illustrate how the offset can be achieved by the TL: command. The first example is for an offset circular reflector and the second example is for an offset square reflector antenna.

EXAMPLE 1

Example 1 is an off-set fed reflector system with a corrugated horn feed designed at NRL [13]. The geometry of the antenna is shown in Figure 1. Two different methods to input the feed information are demonstrated in this example. The first method uses the piecewise linear feed pattern input which was read from the envelope of the measured horn pattern shown in Figure 2.

The second method inputs the feed horn geometry and uses the feedhorn source model to generate the feed patterns in the reflector code. This model eliminates the need to input the feed pattern data point by point. Input is accomplished by specifying the flare angles. The E-plane far field pattern calculated by these two methods are shown in Figure 3 and 4. Note that there is some difference in these two patterns because the first one uses a constant phase feed pattern but the feedhorn model has the phase included. This example also shows the use of the TL: Command for a feed axis tilt. The input data for the first method are given in Table 1 and the input data for the second method are given in Table 2.
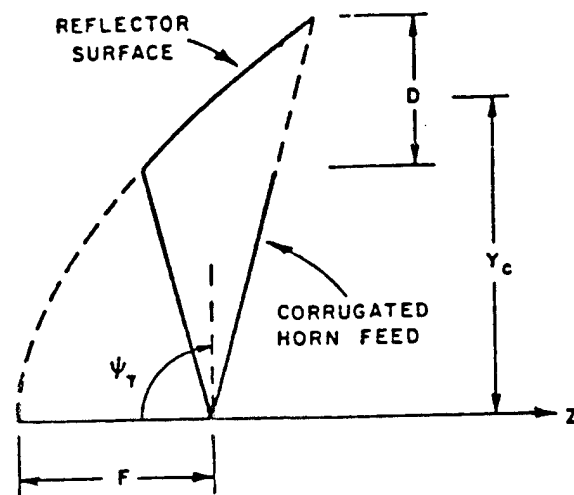
Figure 1. Geometry of an off-set fed reflector antenna with D=35.2 cm, F=32.83 cm, $Y_C$=65.66 cm, $\psi_T$=90°.
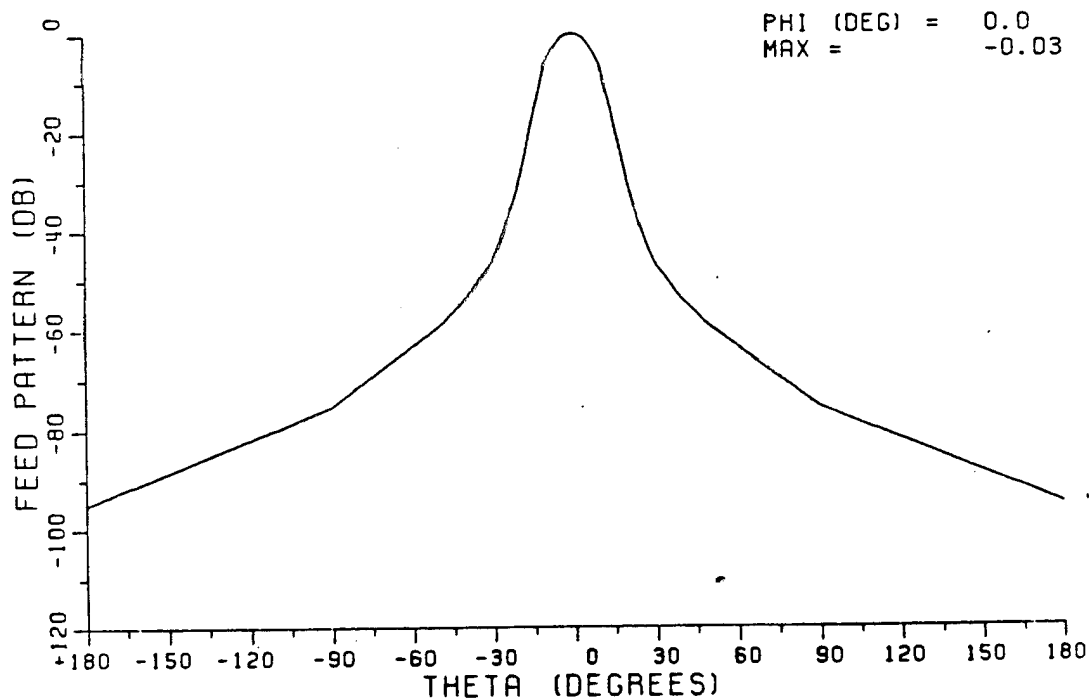


Figure 2. Input feed pattern for offset circular reflector of Figure 1.

C-3
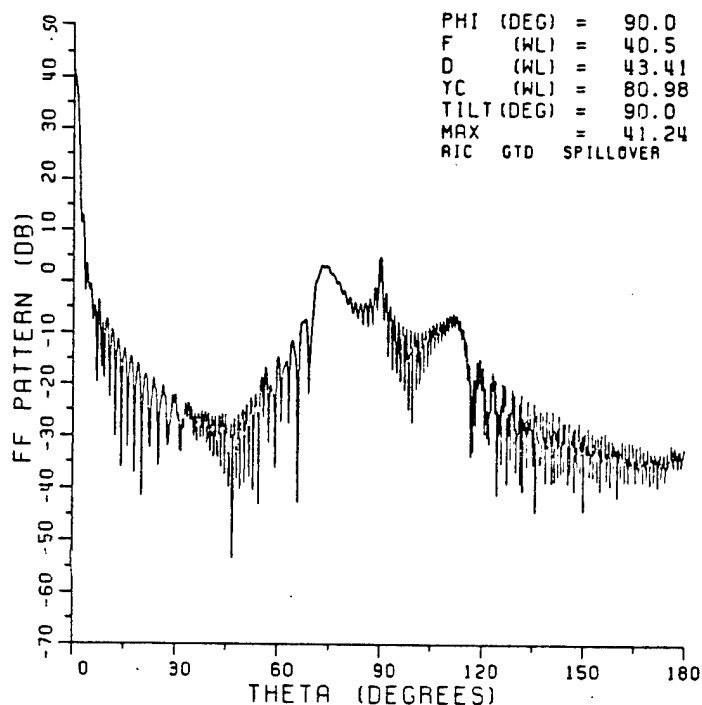
Figure 3. Far field pattern of Example 1 using the piecewise linear feed pattern input. PHI=90.0 degrees.
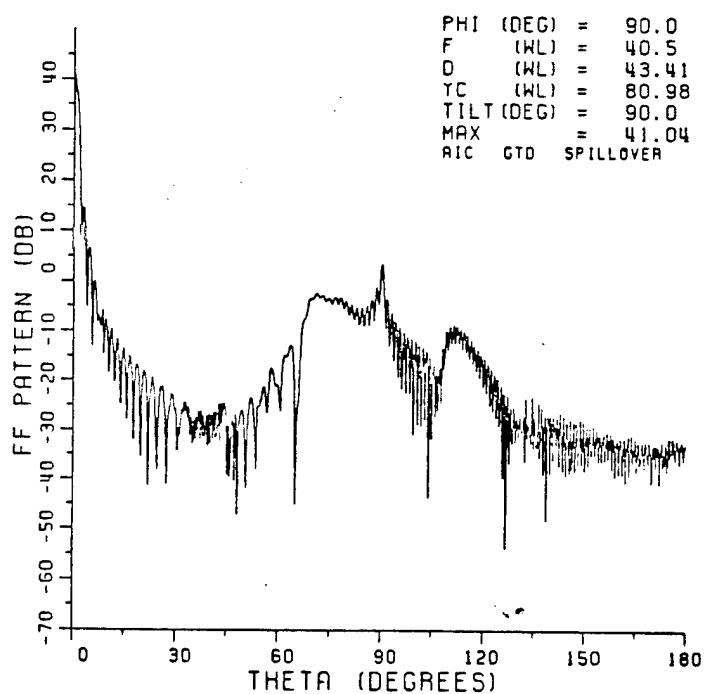


Figure 4. Far field pattern of Example 1 using horn feed geometry inputs. PHI=90.0 degrees.

C-4

## TABLE 1

## INPUT DATA FOR THE OFFSET CIRCULAR REFLECTOR
## WITH PIECEWISE LINEAR FEED INPUT

```
CM:       ***** NRL.DAT *****
CM: EXAMPLE OF OFFSET CIRCULAR REFLECTOR
CE:       LINEAR FEED INPUT
DG:
1
1   0.3281   0.015   0.015   0.352   0
TL:
90.   0.6566
FD:
0   T
T   0   T   1   90.   0.   F
2   0.   90.
1Z
   0.      -0.03    0.    -300.0    0.
   2.      -0.10    0.    -300.0    0.
   5.      -1.20    0.    -300.0    0.
  10.      -6.12    0.    -300.0    0.
  15.     -17.60    0.    -300.0    0.
  20.     -30.10    0.    -300.0    0.
  25.     -39.60    0.    -300.0    0.
  30.     -46.70    0.    -300.0    0.
  40.     -54.00    0.    -300.0    0.
  50.     -59.40    0.    -300.0    0.
  90.     -75.50    0.    -300.0    0.
 180.     -95.00    0.    -300.0    0.
   0.      -0.03    0.    -300.0    0.
   2.      -0.10    0.    -300.0    0.
   5.      -1.20    0.    -300.0    0.
  10.      -6.12    0.    -300.0    0.
  15.     -17.60    0.    -300.0    0.
  20.     -30.10    0.    -300.0    0.
  25.     -39.60    0.    -300.0    0.
  30.     -46.70    0.    -300.0    0.
  40.     -54.00    0.    -300.0    0.
  50.     -59.40    0.    -300.0    0.
  90.     -75.50    0.    -300.0    0.
 180.     -95.00    0.    -300.0    0.
FQ:
1   37.
PZ:
1
90.
0.   180.   0.2
F
PP:
1
1   1
1   2
PF:
1
0.
-180. 180. 2.
1
2
XQ:
```

# TABLE 2

## INPUT DATA FOR THE OFFSET CIRCULAR REFLECTOR
## WITH FEED HORN GEOMETRY INPUT

```
CM:        ***** NRH.DAT *****
CM: EXAMPLE OF OFFSET CIRCULAR REFLECTOR
CE:     FEED HORN GEOMETRY INPUT
DG:
1
1  0.3281  0.015  0.015  0.352  0
TL:
90.  0.6566
FD:
2  T
F .
30.  0.254  30.  0.254  0.476  90.  F
FQ:
1  37.
PZ:
1
90.
0.  180.   0.2
F
PP:
1
1  1
1  2
XQ:
```

EXAMPLE 2

     Example 2 uses the measured data published in Reference [14] for a
square offset reflector antenna. The offset reflector with a 50.8 cm
square aperture and a corrugated horn feed is shown in Figure 5. The
measured feed patterns [14] are shown in Figure 6 for 20.6 GHz, and the
linear feed input which was obtained by extrapolating the measured
pattern is shown in Figure 7. The far field patterns calculated from
the reflector code are shown in Figure 8 and are in excellent agreement
with the published far field patterns shown in Figure 9. The published
near field measurements are shown in Figure 10 for the two principal
plane cut ($\phi=0°$ and $90°$) in the constant Z plane at $Z=1.07$ m. The near
field calculations from the reflector code as shown in Figure 11 agree
very well with the measured results. The input data for the far field
and near field pattern calculations are given in Tables 3 and 4,
respectively.
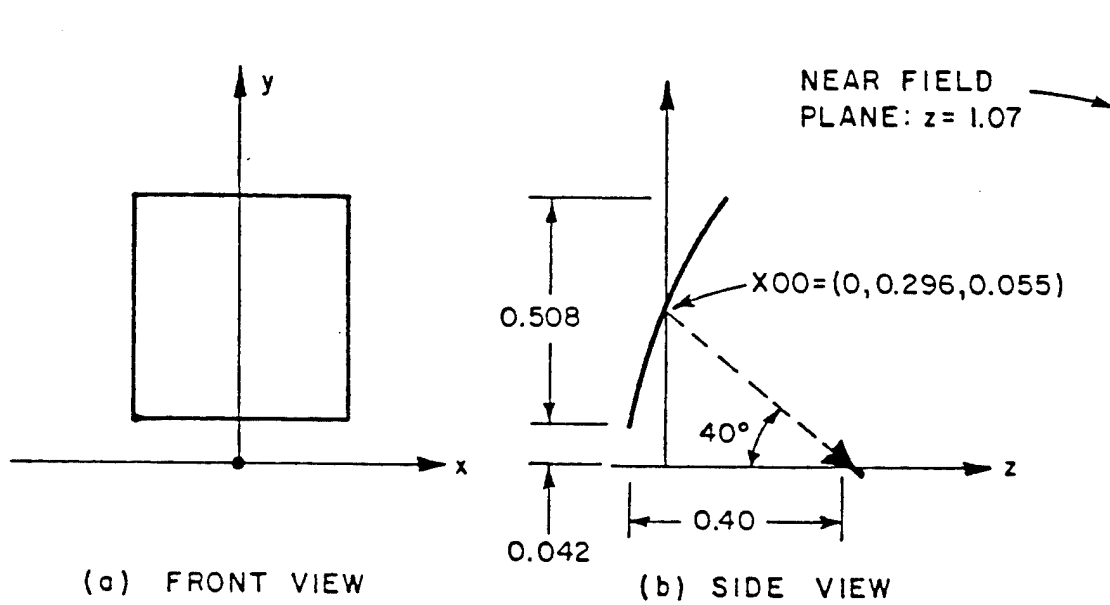
Figure 5. Square reflector of Reference [14]. Dimensions in meters.
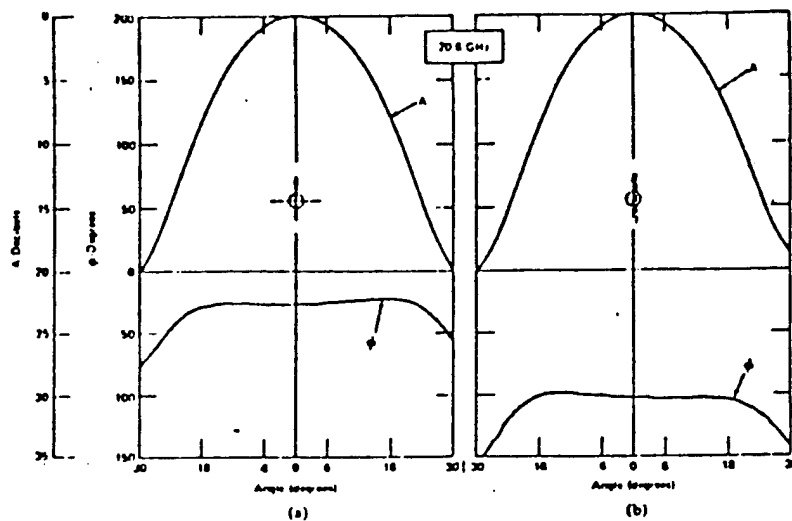
Figure 6. Measured patterns of corrugated horn feed for square reflector [14].
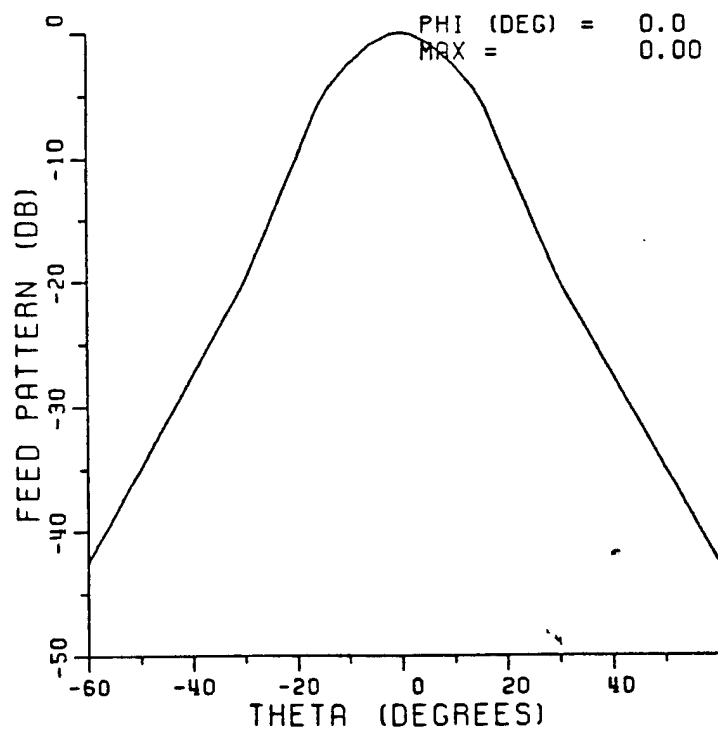


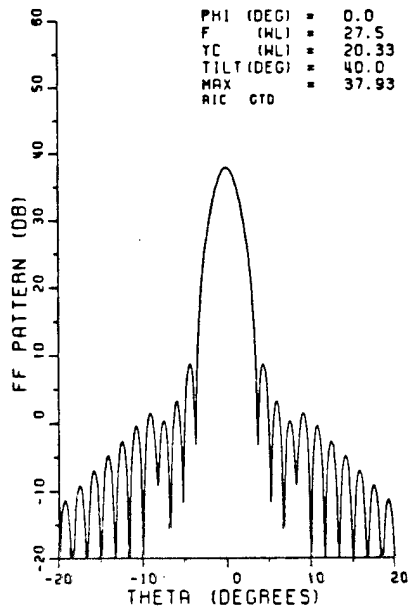Figure 7. Linear feed pattern input for square reflector.

Figure 8a. Far field pattern of
Example B computed by
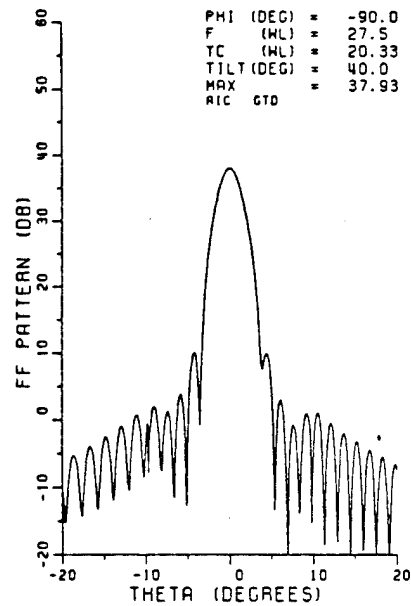OSU Reflector Antenna
Code. PHI=0.0 degrees.

Figure 8b. Far field pattern of
Example B computed by
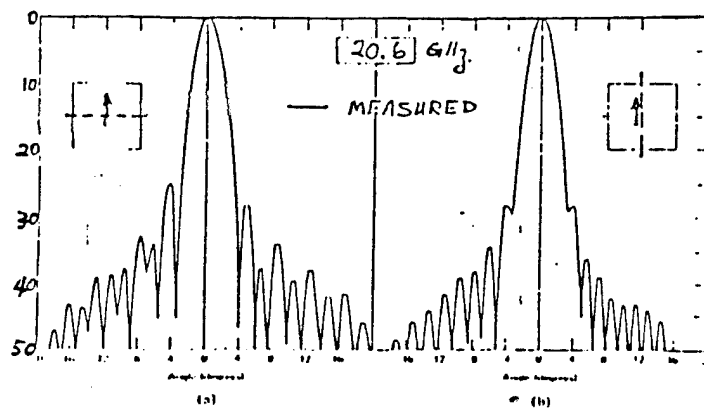NEC Reflector Antenna
Code. PHI=-90.0 degrees.



Figure 9. Far field patterns derived from near field measurements.

## TABLE 3

## INPUT DATA FOR THE FAR FIELD PATTERN CALCULATION OF EXAMPLE B

```
CM:      ***** SQ20L.DAT *****
CM: EXAMPLE OF OFFSET SQUARE REFLECTOR
CE:              FAR-FIELD
DG:
1
1   0.4    0.02    0.02    0.  4
-0.254     0.042
 0.254     0.042
 0.254     0.55
-0.254     0.55
TO:
F   30.  1.
F    0    0    0    0
F    F    F    F    0
T    T    F    0.8
T    F    F    F    T    F    0.  0.
T    T    0.  0.
TL:
40.    0.296
FD:
0    F
0.    0.00816    -0.00967
T    0    T    1    0.  0.  F
1    0.
12
   0.       0.00    -25.    -300.00    0.
   1.      -0.05    -25.    -300.00    0.
   2.      -0.10    -25.    -300.00    0.
   3.      -0.40    -25.    -300.00    0.
   6.      -1.00    -25.    -300.00    0.
  10.      -2.50    -25.    -300.00    0.
  15.      -5.00    -25.    -300.00    0.
  18.      -8.00    -30.    -300.00    0.
  25.     -15.00    -50.    -300.00    0.
  30.     -20.00    -75.    -300.00    0.
  90.     -65.00   -120.    -300.00    0.
 180.    -100.00   -180.    -300.00    0.
FQ:
1    20.6
PZ:
2
0.    -90.
-20.    20.    0.1
F
PP:
1
1    1
1    2
PF:
1
0.
-60.    60.    2.
1
2
XQ:
```

# TABLE 4

## INPUT DATA FOR THE NEAR FIELD PATTERN CALCULATION OF EXAMPLE B

```
CM:       ***** SQ20N.DAT *****
CM: EXAMPLE OF OFFSET SQUARE REFLECTOR
CE:              NEAR-FIELD
DG:
1
1    0.4     0.01     0.01    0.    4
-0.254      0.042
 0.254      0.042
 0.254      0.55
-0.254      0.55
FD:
0   T
T    0    T    1    0.   0.    F
1    0.
12
   0.     0.00   -25.   -300.  0.
   1.    -0.05   -25.   -300.  0.
   2.    -0.10   -25.   -300.  0.
   3.    -0.40   -25.   -300.  0.
   6.    -1.00   -25.   -300.  0.
  10.    -2.50   -25.   -300.  0.
  15.    -5.00   -25.   -300.  0.
  18.    -8.00   -30.   -300.  0.
  25.   -15.00   -50.   -300.  0.
  30.   -20.00   -75.   -300.  0.
  90.   -65.00  -120.   -300.  0.
 180.  -100.00  -180.   -300.  0.
TL:
40.   0.296
FQ:
1    20.6
NF:
0.    0.296   0.055
T
F
0.
PZ:
1
1.07
-0.4  0.4  0.01
F
PP:
1
1   1
1   2
XQ:
NF:
0.    0.296   0.055
T
F
90.
PZ:
1
1.07
-0.4  0.4  0.01
F
PP:
1
1   1
1   2
XQ:
```
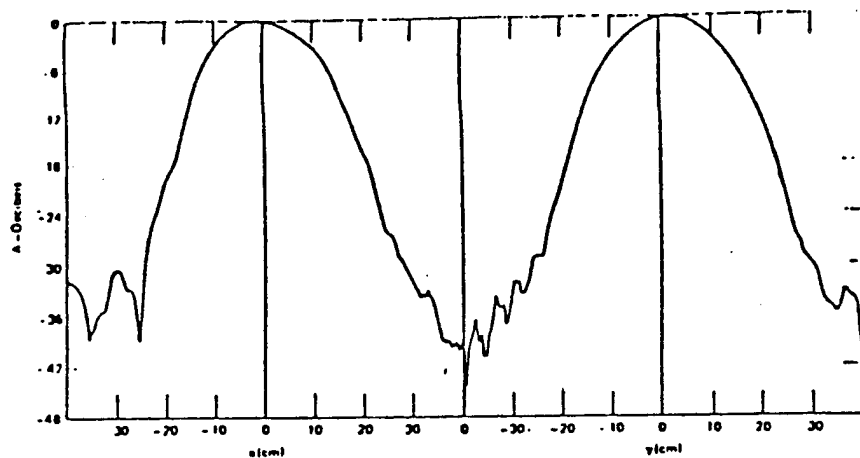
Figure 10.  Measured near field of square reflector.  Frequency=20.6
GHz, z=1.07 meters.
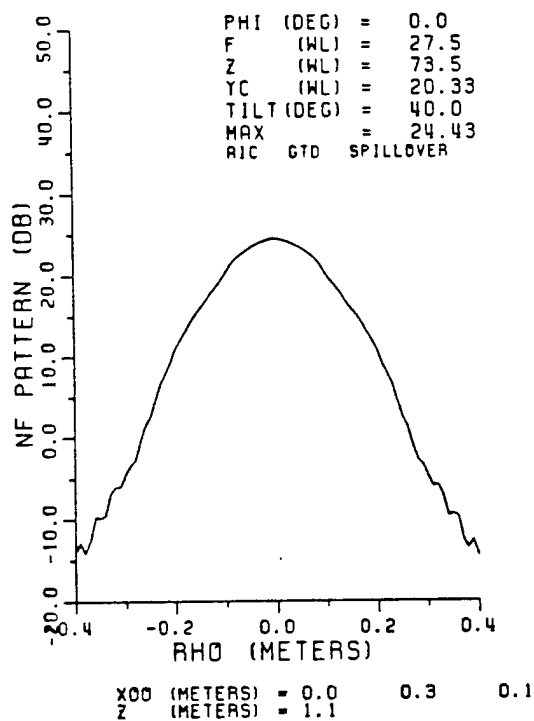


| PHI (DEG) | = | 0.0 |
| F (WL) | = | 27.5 |
| Z (WL) | = | 73.5 |
| YC (WL) | = | 20.33 |
| TILT (DEG) | = | 40.0 |
| MAX | = | 24.43 |

AIC   GTD   SPILLOVER

| XOO (METERS) | = 0.0 | 0.3 | 0.1 |
| Z (METERS) | = 1.1 |

Figure 11a.  Near field pattern
of Example 4 computed
by NEC Reflector
Antenna Code.
PHI=0.0 degrees.

| PHI (DEG) | = | 90.0 |
| F (WL) | = | 27.5 |
| Z (WL) | = | 73.5 |
| YC (WL) | = | 20.33 |
| TILT (DEG) | = | 40.0 |
| MAX | = | 24.47 |

AIC   GTD   SPILLOVER

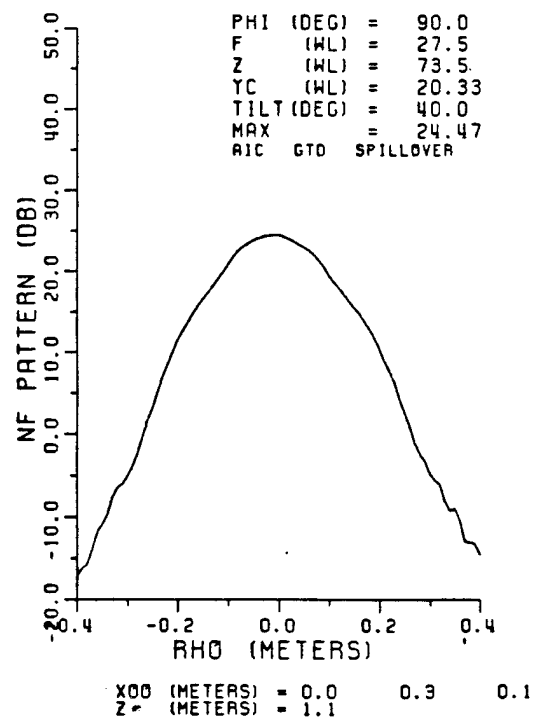| XOO (METERS) | = 0.0 | 0.3 | 0.1 |
| Z (METERS) | = 1.1 |

Figure 11b.  Near field pattern
of Example 4 computed
by NEC Reflector
Antenna Code.
PHI=90.0 degrees.

C-13